Financial Services Data & Analytics Newsletter

March 2024

pwc

Click to launch

Table of contents

Introduction

Globally, we are in the middle of a rapid proliferation of artificial intelligence (AI)-based applications which are solving some of the most complex use cases with unstructured data such as real-time multilingual translation support and product recommendations based on search. For applications that include large language models (LLMs) and semantic search, efficient data processing is more important than ever.

In this newsletter, we will be talking about vector databases, their features as well as their application in the financial services industry. The newsletter also contains industry news from partnership and alliances between industry stalwarts and new-age companies to offer an improved customer experience and optimise efficiencies of the financial services ecosystem.

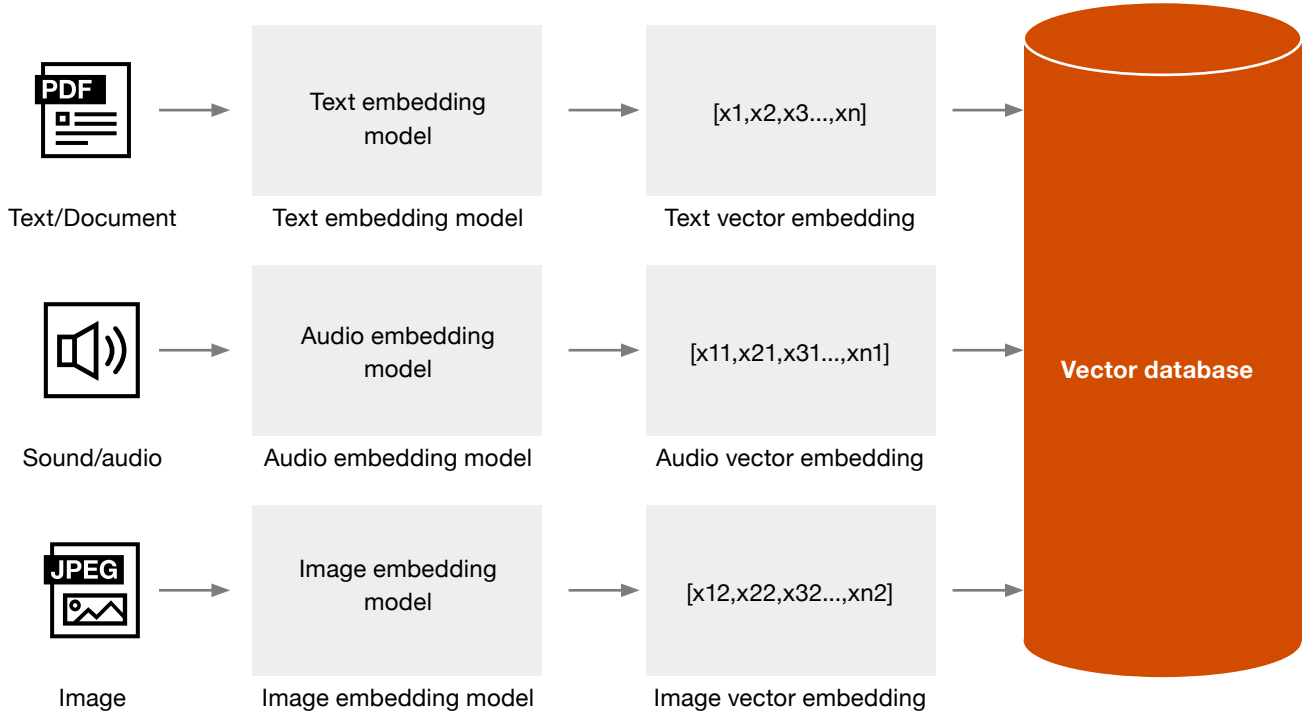
Happy reading!

01 Vector database art of mining unstructured data

With various modern applications generating petabytes of unstructured data, there has been increased focus on harnessing insights from such dataset. Most new applications are all dependent on vector embeddings, which is a kind of vector data representation that contains semantic information essential for AI processing and data maintenance for carrying out challenging tasks. When dealing with unstructured data where contextual knowledge or relationship between entities is important, it can be challenging to extract insights and conduct real-time analysis because typical scalar-based databases are unable to handle the complexity and scale of such data. By leveraging the inherent structure and relationships encoded in vector representations, vector databases facilitate efficient storage, retrieval and analysis of data. Some of the applications/tools like Anuvaad (a judicial domain, document-translation platform to translate judicial documents at scale) and Bhashini (reducing the gap between diverse languages that people speak across the country) have been powered with the same technology. For instance, the likes of Netflix, Spotify work on vector database technology to suggest playlist recommendations to end users based on their search activity. Thus, with the goal of unleashing the power of searching across unstructured data, vector database products have been able to increasingly penetrate into the market.

What are vector databases?

Vector databases are designed to manage and store massive volumes of data as data points in a vector space. They have ability to work with the volatile nature of data and unlock the potential to mine insights from massively unprocessed/untapped data of nature as shown in the infographic below:



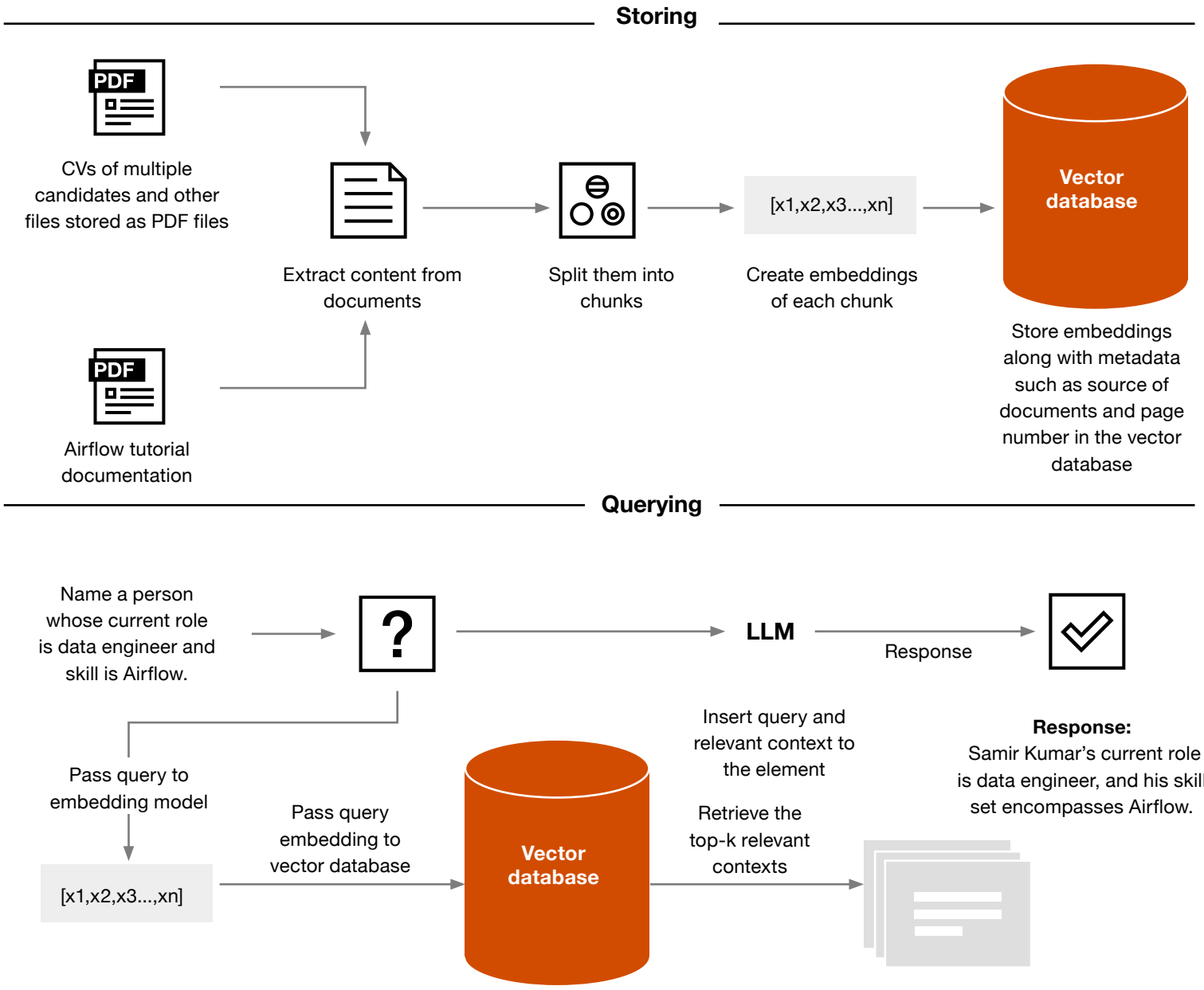
Data from such unstructured data sources goes through a process of tokenisation, conversion into vector embedding, vector storage and retrieval through similarity search algorithms. We can further understand the vector database architecture and its features with a sample use case discussed below.

Architecture and features of vector databases

Vector databases help in the storage of massive volumes of data into the vector space and retrieve data based on querying, fulfilled by techniques like approximate nearest neighbour (ANN) search. With this kind of searching mechanism, they have the ability to return multiple results corresponding to all keys picked up during the search. The following example will help us visualise this concept in detail.

Problem statement: Typically, for any job opening, there are numerous applications received in the form of .pdf and .doc files which need to be filtered and screened based on some defined criteria. This task is almost a very frequent requirement, and the manual process of scanning through each document is laborious and time consuming.

Flow diagram: We will now understand the features and working of vector databases referring to the use case diagram flow as seen here:



1. **Chunking:** The process of transforming large datasets into embeddings is extremely crucial and begins with determining the chunk size, which is the process of breaking down fully extracted data into smaller parts, or 'chunks'. The selected chunking method and parameters help in:
   a. building a suitable context to enable closest match search based on, for example, semantic search.
   b. breaking data into random sizes, which can add a layer of security.

   Example of sentence chunking:

   Sentence_chunking ('Samir's current role is data engineer and his skill set encompasses Airflow. He has good exposure in PowerBI') = ('Samir's current role is data engineer and his skill set encompasses Airflow.', 'He has good exposure in PowerBI')

   The above chunking is performed by assigning a fixed chunk size which will be further used to determine the same number of dimensions in vectors.

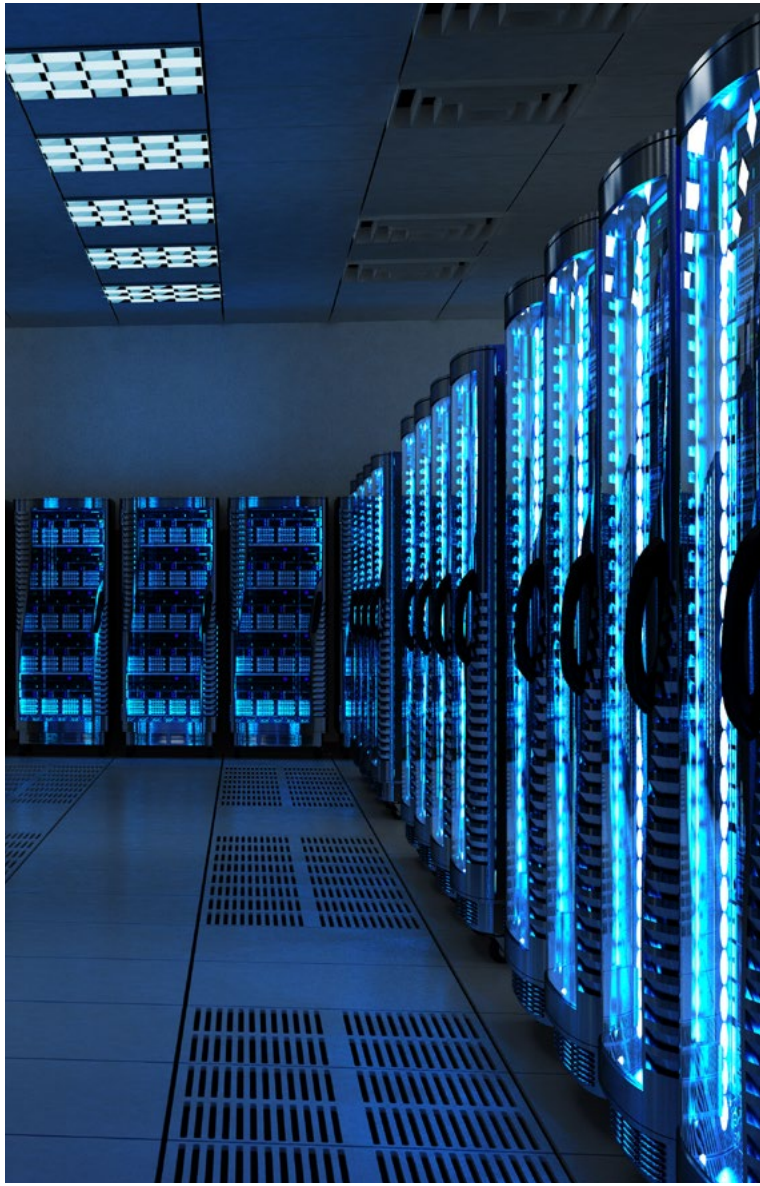
2. **Embedding:** Before storing the data in a vector database, it is necessary to convert raw data into vector embedding. There are various techniques – sentence2vec, speech2vec, R2V, etc. – which help in the conversion of raw data into corresponding vectors in the chosen N dimension space. One of the sample illustrations for the same is shown below:

   Sentence2Vec('Samir's current role is data engineer and his skill set encompasses Airflow.', 'He has good exposure in PowerBI') = ([0.26009 0.4578709 -0.9879877 0.40232 0.5656567 -0.89009], [0.2318969 -0.34156283 0.5610001 0.6768790 -0.7817829 -0.450980898])

   Usually, vector databases come with a default embedding model that can be leveraged. However, it is also possible to use external modules for better compatibility with LLMs to bring out the best results of the target use case.

3. **Storage:** To store any data in vector space and to be able to search with the closest neighbour, the storage components are as below:
   a. **Vector embedding:** ('Samir's current role is data engineer and his skill set encompasses Airflow.') = ([0.26009 0.4578709 -0.9879877 0.40232 0.5656567 -0.89009])
   b. **Metadata:** '//0.0.0.0/Resume_Samir_Kumar.pdf'- This is effectively contextual information which will help in retrieving the original content.
   c. **Vector index:** Local index ID which directly points to vector embedding. This helps in fast retrieval of the data.

4. **Vector indexing :** Any vector database is regarded by its ability to store massive volumes of data and support accurate and prompt vector search operation. To do so, a vector database maps each embedding with a vector index, which then enables it to speed up the search operation. After the initial ingestion, the vector database performs regular index refresh activity which helps accommodate new incoming data into the vector space, thus ensuring re-clustering of data. Each vector database has its own indexing technique which largely determines its scalability and performance.



5. **Retrieval:** In the vector space of N dimension, one needs to specify the query and approximate number of results (k) that must be retrieved in the background. Search operation is performed by converting query into vector embedding and calculating similarity index using methods like cosine similarity or Euclidean distance, which return the results in the order of most relevance. The following example will help us understand this in detail:

Airflow-related data in vector database:

Stored vector data	Metadata location	Storage context
('Apache Airflow is an open source platform for')	//0.0.0.0/Apache_Airflow.pdf	Airflow documentation
('Samir's current role is data engineer and his skill set encompasses Airflow')	//0.0.0.0/Resume_Samir_Kumar.pdf	Candidate with Airflow experience

ANN results count is set to k=2:

Query: What is Apache Airflow			Query: Candidate with Airflow skillset		
EU distance: 0.008	1	('Apache Airflow is an open source platform for')	EU distance: 0.002	1	('Samir's current role is data engineer and his skill set encompasses Airflow')
EU distance: 0.012	2	('Samir's current role is data engineer and his skill set encompasses Airflow')	EU distance: 0.098	2	('Apache Airflow is an open source platform for')

While embedding models are able to retrieve k nearest neighbours against the user query, the most significant task of filtering the appropriate result for the end user lies with LLM. The k retrieved output data is parsed by LLM which is able to generate context against the query and return only meaningful results as shown below.

Query: What is Apache Airflow	**Query:** Candidate with Airflow skillset
Result: //0.0.0.0/Apache_Airflow.pdf	**Result:** //0.0.0.0/Resume_Samir_Kumar.pdf

This is how LLMs powered by vector databases play a pivotal role in enabling the most relevant search operations. This walkthrough is a typical representation of how search operations are performed locally within the limits of trained LLM models. However, to unleash the power of vector search, it may be necessary to access the dynamically updated external knowledge repository which will help in building accurate contexts for the query.

6. Retrieval augmented generation (RAG): RAG is a mechanism of formulating additional related context to user query which will enable an LLM to support efficient vectors even without being trained on context related to query. An example to understand the power of RAG is given below.

Query: List candidate who has rich hands-on experience in Apache Airflow	
With RAG	**Without RAG**
Additional context: Candidate with Airflow skillset	No additional context
Result: //0.0.0.0/Resume_Samir_Kumar.pdf	**Result:** No match identified

The above example shows how RAG could supply additional context 'Candidate with Airflow skill set' against the query including words like 'rich hands-on experience' which were not directly interpretable by the embedding model. Additional context generated with RAG serves as a base for the LLM to interpret the query's context.

Use cases for vector databases

1. Image and video recognition

Vector databases can be used to find similar images, duplicate detection or image categorisation. The high-dimensional nature of the vector database facilitates storing the images and videos content into vector embeddings.

Example: In case of a car accident, the adjuster can simply ask AI to 'show me images similar to this crash' and a vector search-powered system can return photos of car accidents with similar damage profiles from the claims history database.

2. Natural language processing (NLP)

In NLP, words or sentences can be represented as vectors through embeddings. With vector databases, finding semantically similar texts or categorising large volumes of textual data based on similarity becomes feasible, becoming apparent in the semantic analysis.

For example, in a customer support chatbot system, customer queries are transformed into vectors using embeddings. When a user asks, 'How do I reset my password?', the vector database can identify semantically similar queries like 'Steps for password change' to provide a relevant response even if the exact phrasing isn't in the system.

3. Product recommendation

Vector databases are naturally designed to return the most closely related data during a vector search operation. Thus, they can be leveraged for suggesting content/product recommendations based on usage type/search operations fired by the end user.

For example, an online retail customer can be prompted with product recommendations which are bundled together and stored in the vector space. This will enhance the user experience by suggesting most suitable products corresponding to search or purchase history.

4. Media analysis and storage management

In the image and video processing domain, the likelihood of storage overrun is evident due to the similar nature of image/audio files captured. Thus, efficient media analysis and storage management is required to manage the growing volume of files.

For example, similar images, duplicate files stored in the system can be subjected to vector database search for real-time storage management. The application hosted on top of this vector database can suggest cleanup of redundant data files, thus helping in efficient storage management.

Vector database – key parameters

Vector databases are becoming increasingly popular nowadays. Some of the existing databases have started offering extensions to support vector search. There is also a collection of dedicated open-source vector database products that come with their own unique set of offerings. There are other companies which provide proprietary vector databases with full-stack features, including serverless support. While selecting a vector database, the following factors should be taken into consideration:

1. **Ease of installation:** First, we need to install a vector database. To do this, we should consider the difficulty level of installing a vector database and configuring it accordingly. The stack is expected to provide an interactive GUI to manage the database for ease of maintenance.
2. **Cost:** Assess the cost of the licence, underlying hardware, maintenance etc., before selecting a vector database. There are multiple open-source vector databases which can help to reduce the overall cost.
3. **Performance:** Checking on the performance of the database for the size of data within your organisation is very important.
4. **Cloud compatibility:** Keeping up with dynamic changes, most organisations are now moving to cloud computing for scalability. Similarly, while selecting a vector database, we need to check on its cloud compatibility.
5. **Managed vs self-hosted:** Hosting and maintaining the underlying infrastructure can be cumbersome. Also, it would require a dedicated team to manage it. There are multiple options available in the market for managed vector databases which can be selected to reduce these overheads.
6. **Developer experience:** API suite, product support community, error handling etc., should be considered while choosing a vector database. This would help in faster development and quicker issue resolutions.
7. **Indexing capability:** Index management is a key feature of a database that can reduce the response time for end users while querying the database. There are some products which offer real-time indexing without additional overhead to manage it separately. Suitable products must be chosen accordingly, depending on the use case.
8. **Multitenancy:** Vector databases must have the ability to work with namespaces in indexes to ensure infrequently queried namespaces do not increase costs.
9. **Read-after-write consistency:** This is the ability of a vector database to be queryable with the latest data after ingestion of new data.

Conclusion

Vector databases are likely to become a commodity due to the increasing demand of managing unstructured data and deriving insights from them. They provide a way to increase the performance, scale according to the increasing data size and enable flexible use of this stored data for a variety of AI-driven applications. We've highlighted a few important points that will drive the maturity to use vector databases at-scale below:

1. High scalability of vector databases allows us to handle large datasets. However, this has to be achieved at an optimal cost with serverless architectures.
2. The accuracy of the results returned may directly affect the speed of vector search. So an increase in speed may reduce the accuracy. We may need to trade-off between the speed and accuracy by tuning the vector index based on individual requirements.
3. For faster retrieval of the data, indexing plays an important role in vector databases as it stores massive amounts of data. We may need to consider the vector database which has capabilities to perform real-time indexing algorithms. Indexing can also affect the storage. If not indexed efficiently, the indexes may consume a high volume of storage.
4. The vector database search operation is usually tightly coupled with the LLM used for context building and retrieval. It would be a handy option if vector database products have the feature of enabling RAG for accessing external knowledge libraries which acts as an active context builder and search operation accelerator.

5. Every database has different built-in offerings. A vector database for a particular use case must be chosen by evaluating the key features as mentioned in the section above.
6. Security is going to be an important aspect since regulated organisations often resist using open-source LLMs at enterprise scale due to security risks. Hence, it is important to design secured handshakes between the LLM and vector database integration.

Note that vector search operations can be coupled with various machine learning pipelines to further tailor-fit the outcome of a desired use case. In other words, while a vector database has its own set of native offerings that stand out as compared to traditional scalar databases, there is significant scope of enabling advanced vector operations with further enhancements in these products.

01 Reserve Bank of India (RBI) halts Paytm Payments Bank operations

RBI ordered Paytm Payments Bank to cease all banking services within a month, affecting transactions through UPI, IMPS and Aadhaar-enabled payments. This move impacts both users and merchants relying on Paytm for financial services.

02 ICICI Bank's tech-driven transformation yielding remarkable results

ICICI Bank accelerated its journey towards becoming a tech-centric institution, witnessing substantial growth in ROE, net profit and a decrease in net NPAs under CEO Sandeep Batra's leadership. With a strategic focus on leveraging technology, including digital infrastructure and AI-powered services, the bank has garnered accolades such as being named 'Bank of the Year' in the BT-KPMG Best Banks and FinTechs survey for three consecutive years.

03 RBI instructed credit bureaus to improve data quality

RBI has asked credit information companies to improve data quality, focus on timely redressal of customer complaints and revamp the process of handling customer requests.

04 GST Network to share business data with RBI's tech platform for faster credit access

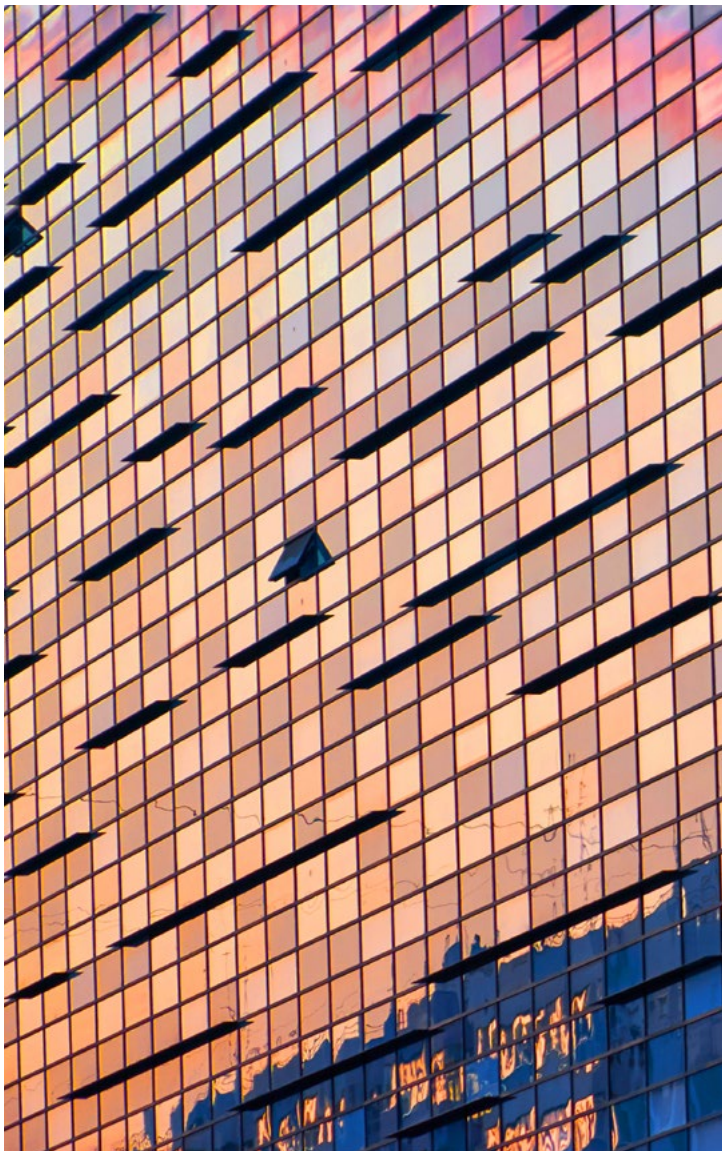
The Government has notified the GST Network to share GST-registered businesses data with RBI's Public Tech Platform for Frictionless Credit. This will help businesses to get faster loans for various credit needs.

05 Treasury department recovers over USD 375 million with new AI-enhanced fraud detection system

The Department of the Treasury, on 28 February 2024, announced the implementation of an AI-enhanced fraud detection system designed to tackle cheque fraud in near real-time. This innovative approach has significantly bolstered the department's ability to recover potentially fraudulent payments from financial institutions, leading to the recovery of more than USD 375 million.

06 ASEAN enhances AI governance and ethics guide amid regulatory uncertainty

The Association of Southeast Asian Nations (ASEAN) recently approved an AI governance and ethics guide during their meeting in Singapore. The guide aims to standardise AI frameworks among its member countries, despite the region's varied regulatory approach to AI and generative AI technologies.



07 Canara Bank inaugurated its cutting-edge DnA centre in Bangalore

Canara Bank launched its Data and Analytics (DnA) Centre in Bangalore, equipped with AI and machine learning capabilities. The centre aims to enhance operational efficiency and customer experience, and the bank plans to redefine banking services for customers with new products and training modules. The launch event also included a hackathon showcasing innovative banking solutions, highlighting the bank's focus on leveraging technology to meet the evolving needs of the banking sector and customers.

08 The BFSI sector embracing AI upskilling path to stay competitive amidst the generative AI shift

The banking and financial services industry (BFSI) is implementing artificial intelligence (AI) to enhance its operations and customer service. Industry leaders are emphasising the importance of upskilling the workforce in AI technologies, such as machine learning and natural language processing, to remain competitive in the generative AI era.

09 CARS24 Financial Services collaborated with Credgenics to digitise collection processes

CARS24 Financial Services, a car financing and wholly owned subsidiary of CARS24, has announced a strategic partnership with Credgenics, a SaaS-based debt collection and resolution technology platform. It will help automate loan-related customer communications across channels for faster collections.

10 Bajaj Allianz has launched a generative AI bot, Insurance Samjho

Bajaj Allianz has launched a generative AI-based bot called 'Insurance Samjho'. Customers may access the website, where they are asked to upload the documents. The bot acts as a virtual assistant, providing answers in a conversational style.

Contact us

Mukesh Deshpande
Partner, Technology Consulting
mukesh.deshpande@pwc.com

Hetal Shah
Partner, Technology Consulting
hetal.d.shah@pwc.com

Acknowledgements

This newsletter has been researched and authored by Akash Tiwari, Antima Garg, Arpita Shrivastava, Fenil Thakkar, Garima Yadav, Mihir Shah, Raghav Sharma, Rahul Chellani, Ritvik Hariharan, Neeraj Sibal, Shivangi Mitra and Shruti Agrawal.

About PwC

At PwC, our purpose is to build trust in society and solve important problems. We're a network of firms in 151 countries with over 360,000 people who are committed to delivering quality in assurance, advisory and tax services. Find out more and tell us what matters to you by visiting us at www.pwc.com.

PwC refers to the PwC network and/or one or more of its member firms, each of which is a separate legal entity. Please see www.pwc.com/structure for further details.

pwc.in

Data Classification: DC0 (Public)

HS/April 2024 - M&C 36662