© Volume 2, Issue 10, October 2026 | JATIR
e-ISSN: 2839-5786, p-ISSN: 3139-8464

Multilingual and Multimodal Business Intelligence Agent: A Voice Driven Analytics System for Handwritten, Printed and Digital Business Records

[1]Sayali Gaikwad, [2]Omkar Shelke, [3]Atharva Rasal, [4]Shubham Chinchkar
[5]Om Patil

[1]Assistant Professor Department of Information Technology Parvatibai Genba Moze College of Engineering, Wagholi, Pune, India

[2,3,4,5]Student Department of Information Technology Parvatibai Genba Moze College of Engineering, Wagholi, Pune, India

***Abstract*—Small and medium businesses generate valuable transactional data, yet much of it remains locked in handwritten registers, paper bills and phone photographs, and the analytics tools that could exploit it presuppose structured data, technical skill and English language interaction. Recent advances in transformer based optical character recognition (OCR), Indian language speech recognition, large language model (LLM) agents, retrieval augmented generation (RAG), timeseries forecasting and anomaly detection make it plausible to remove these barriers, but the techniques have largely been studied in isolation. This paper reviews seven enabling technology areas—document digitization, multilingual and codeswitched speech, L-driver text to code analytics, forecasting and anomaly detection, retrieval and memory, data validation and explainability, and voice driven inventory automation—and compares representative methods against the needs of nontechnical Indian shop owners. Building on this review, we propose MMBIA, a layered reference architecture for a Multilingual and Multimodal Business Intelligence Agent that integrates sixteen functional modules around a validation first, human in the loop, local first design, together with an evaluation framework and a catalogue of open research problems. The architecture is a design proposal; no empirical performance claims are made.**

***Index Terms*—business intelligence, multimodal AI, handwritten text recognition, multilingual speech recognition, LLM agents, retrieval augmented generation, demand forecasting, MSME analytics.**

I. INTRODUCTION

Sales, purchases, stock movements, expenses and invoices are the daily byproducts of every small business. Business intelligence (BI) tools promise to turn such records into decisions, but they were designed around three assumptions: that data already exists in a clean digital form, that the user can operate spreadsheet, SQL or dashboard configuration tools, and that the interface language is English. Each assumption fails for a large segment of potential users. Neighborhood retailers such as Kirana stores often keep accounts in handwritten registers or on printed bills, may be more comfortable in Marathi, Hindi or a Hindi–English mixture ("Hinglish") than in English, and cannot easily justify enterprise analytics subscriptions.

At the same time, the building blocks for a more inclusive alternative have matured quickly. Transformer encoder–decoder models read printed and handwritten text from images
[1]; speech models trained on large multilingual corpora and
adapted to Indian languages transcribe spoken queries [2],
[3]; LLMs can write executable analysis code from natural language requests [4], [5]; RAG grounds answers in curated knowledge [6]; and open weight models can run locally [7], [8]. What is missing is a systematic view of how these components behave when chained together for this user population, where handwriting is variable, speech is codeswitched, histories are short and mistakes are costly.

This paper makes the following contributions:

- a structured review of seven technology areas that jointly determine whether a multilingual, multimodal BI agent is feasible, organized around five user facing barriers (Section III);
- a comparative analysis of representative methods and their limitations for small business settings (Section V);
- MMBIA, a layered reference architecture with sixteen modules and a query routing workflow (Section VI);
- an end-to-end evaluation framework with proposed metrics for each layer (Section VII); and
- a catalogue of open research challenges and future directions (Sections VIII and IX).

The remainder of the paper is organized as follows. Section II describes the scope and method of the review. Section III characterizes the problem landscape. Section IV reviews the literature by technology area. Sections V–VII present the comparison, the proposed architecture and the evaluation framework. Sections VIII–X discuss challenges, future work and conclusions.

II. REVIEW SCOPE AND METHOD

This is a narrative, thematic review rather than a systematic literature review. The themes were derived top down from the barriers faced by the target users (Table I), and for each theme we selected (i) foundational method papers, (ii) benchmark or survey papers that characterize the state of the art and its failure modes, and (iii) the peer reviewed or archival descriptions of the specific opensource components considered for implementation. Because LLM agents and small

open weight models evolve faster than journal cycles, several sources are arXiv preprints; these are marked as such in the reference list. Proprietary products without a
published method were excluded. The review is therefore selective, and coverage of the most recent agent frameworks may be incomplete.

The target user population comprises small retailers and analysts working with spreadsheets, invoices, photographed registers or SQL data, who require voice first interaction in Marathi, Hindi, Hinglish or English. The technology baseline considered is a zerolicencecost, opensource stack deployable on free or low-cost hosting and a commodity development machine.

III. PROBLEM LANDSCAPE

Existing analytics workflows can be characterized by five barriers: language (English oriented interaction), data format (handwritten or scanned records instead of spreadsheets), cost (enterprise grade analytics and scheduled reporting), technical skill (spreadsheets, SQL, dashboard configuration) and limited intelligence (descriptive reporting without forecasts, alerts or explanations). Two further concerns arise once automation is introduced: trust, because errors introduced early in a pipeline can silently propagate into confident sounding insights, and context, because owners ask general business questions that no uploaded dataset can answer. Table I maps each barrier to the technology area that addresses it.

TABLE I BARRIERS AND ENABLING TECHNOLOGY AREAS

Barrier	Consequence for the user	Enabling area (Section)
Language	Queries and answers only in English	Multilingual / codeswitched ASR and TTS (IVB)
Data format	Registers, bills and photographs cannot be analyzed directly	Handwritten and printed OCR; document parsing (IVA)
Technical skill	Requires Excel, SQL or dashboard setup	LLM agents with texttocode (IVC)
Limited intelligence	Reports describe only the past	Forecasting; anomaly detection (IVD)
Missing context	General business questions fall outside the data	RAG and memory (IVE)
Trust	Input errors propagate unnoticed	Validation; explainability (IVF)
Cost	Enterprise tools hard to justify	Opensource, localfirst stack (VI)

IV. REVIEW OF ENABLING TECHNOLOGIES

A. Document Digitization and Handwritten Text Recognition

Classical opensource OCR engines such as Tesseract combine layout analysis with line and word level recognizers and perform well on clean printed text [9]. Their accuracy degrades on photographs with uneven lighting, skew or handwriting, which motivates image preprocessing (denoising, deskewing, binarization) with libraries such as OpenCV before recognition.

Transformer based recognizers have shifted the state of the art: TrOCR pairs a pretrained image Transformer encoder with a pretrained text Transformer decoder and reports strong results on printed, handwritten and scene text benchmarks [1], building on the Transformer architecture [10]. Layout aware pretraining such as LayoutLM [11] and OCRfree approaches such as Donut [12] extend recognition to document understanding, i.e. extracting fields and tables rather than raw text.

Three issues make handwritten shop ledgers harder than the benchmarks suggest. First, TrOCR operates on cropped text lines, so a ledger page must first be segmented into lines and cells, and its tabular structure reconstructed afterwards. Second, standard handwriting benchmarks such as IAM are English sentence databases [13]; models must be finetuned for Devanagari script, mixed numerals and shop specific abbreviations. Third, a single misread digit changes a financial figure without producing an obviously malformed output. Practical systems therefore need per field confidence scores, arithmetic consistency checks (for example, line items against stated totals) and a user confirmation step for low confidence values.

B.Multilingual and Code-switched Speech

Largescale weakly supervised training produced speech recognizers that are robust across many languages and accents [2]. For Indian languages, the Vistaar benchmark suite and training sets were used to finetune Whisper models, released as Indic Whisper, and the authors report the lowest word error rate on most of the 59 benchmarks spanning 12 languages [3]. Wider efforts on Indic corpora and evaluation [14], [15] and the sobering analysis of how unevenly NLP resources are distributed across the world's languages [16] explain why such language specific adaptation matters.

Business speech nevertheless remains difficult. Codeswitching—alternating between languages within a sentence, as in Hinglish—is a long recognised challenge for both recognition and downstream language understanding [17], and has dedicated evaluation benchmarks [18]. Additional complications include inconsistent output script (Devanagari versus Romanized text), out of vocabulary product and brand names, units and quantities, and acoustic noise in shops. Spoken replies require text speak engines such as gTTS or Coqui TTS, whose coverage and naturalness for Marathi should be verified rather than assumed.

C.LLM Agents and TexttoCode Analytics

Foundation models [19] trained on code can translate natural language requests into programs [4]. Program aided reasoning delegates calculation to an interpreter instead of the model [5], and agent patterns interleave reasoning with tool calls [20], [21]. In BI terms this means a planner interprets the user's question, a code generator emits Pandas operations [22], and a sandbox executes them and returns tables or figures. Text to SQL benchmarks such as Spider
[23] and BIRD [24] show that schema grounding, ambiguous questions and "dirty" data values remain significant sources of error even for strong models; analogous failure modes are expected for spreadsheet analysis.

For privacy and cost reasons a local deployment of open weight models such as Llama 3 [7] or Gemma 2 [8] is attractive, and quantization-based techniques reduce memory demands [25]. The tradeoff is that smaller models are more prone to hallucination [26] and to malformed code. Two safety issues deserve explicit design attention: generated code must run in a restricted sandbox, and text extracted by OCR or retrieved from documents is untrusted input that could carry instructions (prompt injection) and must never be allowed to alter tool permissions.

D. Forecasting and Anomaly Detection

Prophet fits a decomposable model of trend, seasonality and holiday effects, exposes interpretable components and supports analyst in the loop adjustment [27]. It is attractive for retail because custom event calendars (for example, festival periods) can be added as holiday regressors. However, the forecasting literature cautions that more complex models do not automatically outperform simple ones: the M4 competition found statistical and hybrid approaches highly competitive [28], and surveys of deep forecasting stress its dependence on data volume [29]. Textbook practice recommends benchmarking against naive and seasonal naive baselines and reporting prediction intervals [30]. These cautions are acute for a kirana shop, whose history may span only a few months and include intermittent demand.

For anomaly detection, Isolation Forest isolates unusual observations through random partitioning and scales linearly with sample size [31]; it is available in scikit learn [32]. The broader anomaly detection survey [33] reminds us that “anomalous” is context dependent: a sales spike on a festival day is expected, whereas the same spike on an ordinary day may signal a data entry error. Alerts should therefore carry a severity level and a plain language rationale, and should be suppressed or annotated when a known event explains them.

E. Retrieval Augmented Knowledge and Memory

RAG combines a retriever over an external corpus with a generator, improving factual grounding for knowledge intensive queries [6]. Surveys catalogue the retrieval, augmentation and generation design space [34], while practitioner studies identify recurring concerns for production chatbots—freshness, security, scalability and trust [35]. For an Indian MSME knowledge assistant, the corpus must be curated and versioned because regulations, tax rules and scheme eligibility change; stale answers are a risk even when retrieval works. Queries arrive in Marathi or Hindi while source documents may be in English, so multilingual embedding and answer language control are required.

Conversational and persistent memory allow follow up questions such as “and compared with last month?” without restating context. Mem0 proposes extracting, consolidating and retrieving salient facts from dialogue as long term memory for agents [36]. For BI, memory must additionally hold business profiles, preferences and prior analyses, and must be user inspectable and deletable, since it contains commercially sensitive data.

F. Data Validation and Explainability

Automated data quality verification encodes expectations—completeness, uniqueness, value ranges—as declarative constraints that are checked before downstream use [37]; the opensource Great Expectations library follows this pattern. Validation is particularly important when data originate from OCR, where errors are random rather than systematic. Explainability for tabular models has been addressed by attribution methods such as LIME [38] and SHAP [39]; for BI agents, however, the dominant need is narrative explanation—what changed, why it may have happened and what action could be considered. The central requirement is that every numerical statement in the narrative be traceable to a computed result; otherwise, a fluent explanation can mask a hallucination [26].

G. Voice Driven Inventory and Workflow Automation

Voice based stock management converts an utterance such as "add fifty packets of rice" into a structured intent (item, quantity, unit, action) and updates a database, followed by rule based low stock alerts and reorder suggestions. Workflow engines such as n8n can schedule periodic business health reports and deliver notifications. We found little peer reviewed work evaluating such voice to inventory pipelines for small retailers; the area is mostly documented in tool documentation and practitioner material, which we treat as a gap rather than as evidence. Because a misheard quantity silently corrupts stock levels, a readback and confirm step before committing any change is a minimum safeguard.

V. COMPARATIVE ANALYSIS

Conventional spreadsheet and BI suites excel at interactive exploration once data are structured but leave ingestion, language adaptation and chart configuration to the user. General purpose chat with data assistants reduce the skill barrier but typically assume typed English queries and structured files. The approach reviewed here targets the intersection that neither covers: heterogeneous inputs, Indian language voice, and proactive intelligence. Table II compares the representative components considered for such a system.

TABLE II REPRESENTATIVE COMPONENTS AND LIMITATIONS

Area	Method	Strength	Limitation for this use
Printed OCR	Tesseract + OpenCV [9]	Mature, lightweight, free	Weak on skew, glare, handwriting
Handwritten OCR	TrOCR [1]	Strong linelevel recognition	Needs line segmentation and Indic finetuning
Speech	IndicWhisper [3]	Indian language accuracy	Codeswitching, noise, script variance
Agent	LangChain + local LLM [20], [7]	Private, no percall cost	Hallucination, hardware demand
Forecast	Prophet [27]	Interpretable, event regressors	Needs adequate history
Anomaly	Isolation Forest [31]	Unsupervised, fast	Contextblind without event data

Knowledge	RAG with vector store [6]	Grounded answers	Corpus staleness; cross lingual retrieval
Memory	Mem0 [36]	Cross session continuity	Privacy and consent management
Validation	Declarative rules [37]	Early error detection	Rules need domain tuning

Two observations follow. First, errors compound: an OCR digit error passes through validation, analysis and explanation, so confidence must be propagated along the pipeline rather than discarded at each stage. Second, no single component is the weakest link in all cases; the weakest link depends on the input modality, which argues for modular, swappable components and for routing decisions based on measured confidence.

VI. PROPOSED REFERENCE ARCHITECTURE (MMBIA)

A. Overview

MMBIA is a layered architecture (Fig. 1). A Flutter Android client and a Stream lit web interface communicate with a FastAPI backend, which routes requests to ingestion, voice, validation, agent, analytics, knowledge, reporting and automation components. Persistent storage (SQLite for development, PostgreSQL for deployment) holds users, data sources, documents, processed data, voice queries, analyses, predictions, dashboards, responses, workflows, agent execution logs and a registry of models. The logging of generated code and model versions makes each answer reproducible and auditable.

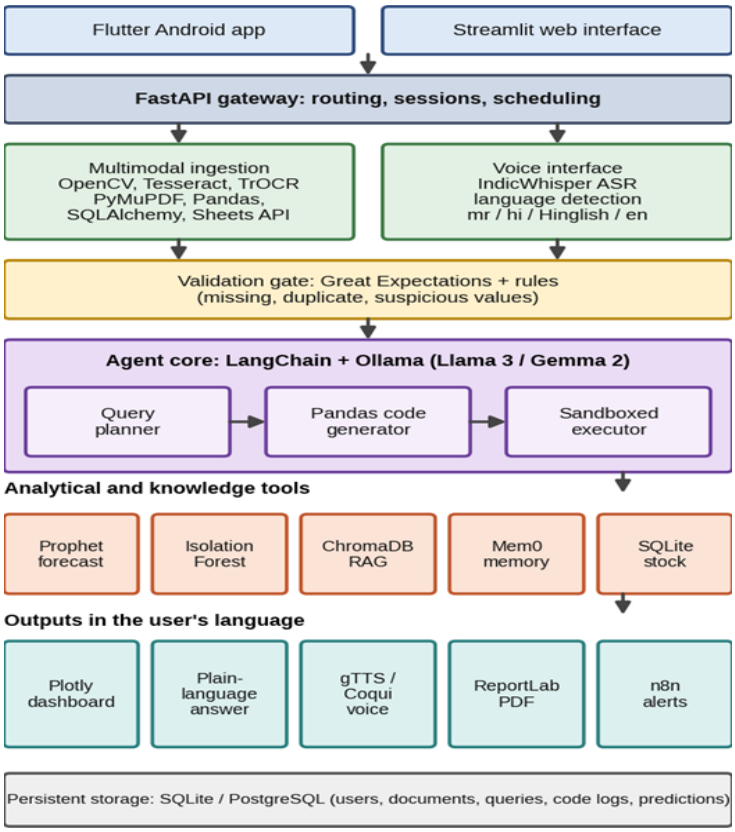


Fig. 1. Layered reference architecture of the proposed MMBIA system.

B.Functional Modules

The sixteen modules are grouped into six functional groups in Table III. The ingestion module routes each input by type: photographs pass through OpenCV preprocessing and Tesseract or TrOCR, PDFs through PyMuPDF, and structured sources through Pandas, SQLAlchemy and the Google Sheets API. Voice input is transcribed by IndicWhisper with language detection. A LangChain agent backed by a local Ollama served model plans the analysis, generates Pandas code and executes it in a sandbox; Plotlybased dashboards present revenue, product, trend, anomaly and profit/loss views with filtering and drilldown. Forecasts use Prophet and scikitlearn with confidence intervals, anomalies use Isolation Forest, and ChromaDB serves the curated MSME knowledge base. ReportLab produces PDF reports and n8n schedules weekly health reports and optional notifications.

TABLE III MMBIA MODULES BY FUNCTIONAL GROUP

Group	Modules
Ingestion and validation	Multimodal document ingestion; smart data validation
Voice and accessibility	Vernacular voice interface; voice driven navigation and accessibility
Analytics	Autonomous text to code analysis; automatic dashboard generation; predictive forecasting; proactive anomaly alerts
Knowledge and memory	Conversational memory; multisession persistent memory; RAG business knowledge assistant; smart question suggestions
Explanation and reporting	Explainable answers; multilingual PDF reports; automated weekly business health report
Operations	Voice based stock management with low stock alerts and reorder suggestions

C.Query Routing Workflow

Fig. 2 shows how a query flows through the system. After transcription and language detection, the agent loads the validated dataset and session context from memory and classifies the intent as descriptive, predictive, knowledge seeking or an inventory command. Descriptive requests invoke generated Pandas code; predictive requests invoke Prophet or Isolation Forest; knowledge requests invoke retrieval; inventory commands are parsed into structured actions. Results are visualized, explained in plain language, optionally spoken, and written back to memory. Smart question suggestions, derived from the available data fields, help users who do not know what to ask.

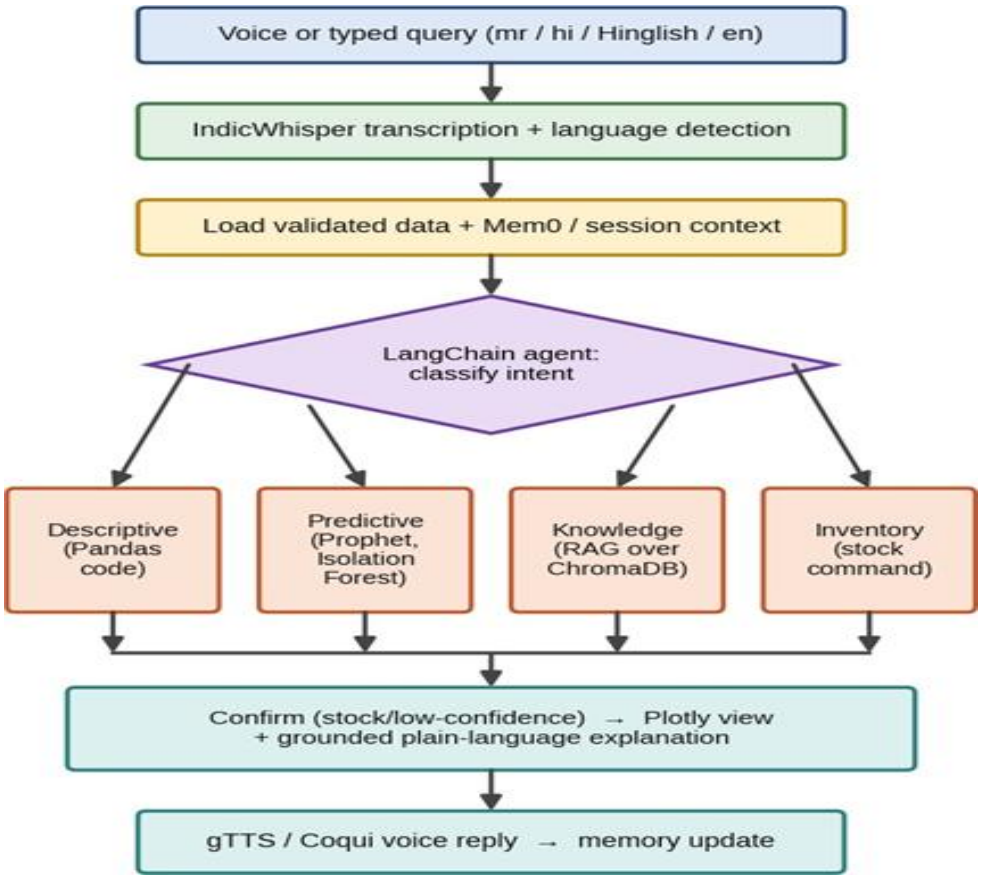


Fig. 2. Query routing workflow from voice input to spoken answer.

D. Design Principles

- Validate before analyzing. No dashboard or forecast is produced until missing, duplicate, incomplete or suspicious values have been surfaced to the user.
- Human in the loop where errors are costly. Low confidence OCR values and every stock change require confirmation.
- Grounded explanation. Narrative text must cite computed figures; forecasts are shown with intervals and framed as decision support, not certainty.
- Local first and modular. Sensitive business data are processed locally where feasible, and each component can be replaced as better models appear.
- Honest about constraints. A zero cost deployment relies on free tiers of hosting and a free GPU runtime for training; these limit memory and uptime, so heavy models may need to run on the developer's own machine, and some integrations are optional.

VII. PROPOSED EVALUATION FRAMEWORK

Component benchmarks alone do not predict end user value, because errors interact. We therefore propose layer wise and end to end evaluation (Table IV). All metrics below are proposed for the future prototype; none has yet been measured.

TABLE IV PROPOSED METRICS BY LAYER

Layer	Proposed metrics
OCR	Character and word error rate; numeric field accuracy; table structure accuracy; calibration of confidence scores
Speech	Word error rate on Marathi, Hindi, Hinglish and English; intent accuracy on codeswitched business commands; robustness to noise
Agent	Answer correctness against hand computed ground truth; code execution success rate; sandbox policy violations
Forecast	MAPE/sMAPE or MASE against naive and seasonal naive baselines; prediction interval coverage
Anomaly	Precision, recall and F1 on injected and labelled anomalies; false alarms on known events
RAG	Recall@k of retrieval; faithfulness of answers to retrieved passages; cross lingual retrieval accuracy
Validation	Detection rate for injected missing, duplicate and out of range values
Inventory	Command to action accuracy; confirmation prevented errors
End to end	Task success and completion time with real shop owners; usability score; latency on low bandwidth links

Evaluation data should include photographed ledgers of varied quality, recorded commands in several dialects and noise conditions, and sales series of different lengths. Where real shop data cannot be shared, consented or synthetic datasets should be documented transparently.

VIII. OPEN CHALLENGES AND RESEARCH GAPS

- Data scarcity. Public datasets of handwritten Indic script business ledgers and of codeswitched, domain specific shop speech are scarce, limiting finetuning and fair comparison.
- Error propagation. Few studies measure how OCR, ASR and LLM errors compound across a pipeline or how uncertainty can be carried to the final answer.
- Reliability of small local LLMs. Text to code accuracy on messy, user specific spreadsheets under hardware limits is underexplored, as are guardrails against hallucinated numbers [26].
- Forecasting with short histories. Methods that remain calibrated with months rather than years of data, and that handle intermittent demand and festival effects, need evaluation in small retail settings [28], [30].
- Security and privacy. Sandboxed execution, prompt injection resistance for OCR derived text, and user-controlled memory deletion require explicit design and testing.
- Trust and explanation. It is unclear which explanation styles nontechnical owners understand and act on; user studies are needed.
- Connectivity and cost. Reliance on cloud hosted components conflicts with intermittent connectivity and free tier limits.

- Fairness across dialects. Recognition accuracy may vary across regional Marathi and Hindi dialects and speaker groups [16].

IX. FUTURE DIRECTIONS

Near term extensions include a fully offline Android version using quantized local models [25], support for additional Indian languages such as Tamil, Telugu, Bengali and Gujarati, and WhatsApp based interaction in which users forward bill photographs and receive analysis. Longer term integrations include GST assistance, import from accounting software such as Tally and Busy, hyperlocal competitor price monitoring, privacy preserving federated market intelligence, and deployment through government and digital service channels such as MahaIT and CSC centres. Each extension raises the evaluation, privacy and data governance questions identified above, and should be pursued only alongside the corresponding measurements.

X. CONCLUSION

This review examined the technologies required to build a multilingual, multimodal BI agent for small businesses and found that each has reached a level of maturity that makes integration realistic, yet each carries failure modes that are amplified when chained together and used by nontechnical people in low resource conditions. The proposed MMBIA architecture responds with a validation first, confirmation based, grounded and modular design, and the accompanying evaluation framework and challenge catalogue identify what must be measured next. Implementing the prototype and reporting empirical results on OCR, speech, analysis, forecasting and usability with real shop data is the logical continuation of this work.

ACKNOWLEDGMENT

The authors thank the Department of Information Technology, Parvatibai Genba Moze College of Engineering, Wagholi, Pune, for support during the preparation of this work.

REFERENCES

[1] M. Li, T. Lv, J. Chen, L. Cui, Y. Lu, D. Florencio, C. Zhang, Z. Li, and F. Wei, “TrOCR: Transformer based optical character recognition with pretrained models,” Proc. AAAI Conf. Artif. Intell., vol. 37, no. 11, pp. 13094–13102, 2023, doi: 10.1609/aaai.v37i11.26538.

[2] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, “Robust speech recognition via largescale weak supervision,” in Proc. Int. Conf. Mach. Learn. (ICML), 2023.

[3] K. S. Bhogale, S. Sundaresan, A. Raman, T. Javed, M. M. Khapra, and P. Kumar, “Vistaar: Diverse benchmarks and training sets for Indian language ASR,” arXiv:2305.15386, 2023.

[4] M. Chen et al., "Evaluating large language models trained on code," arXiv:2107.03374, 2021.
[5] L. Gao et al., "PAL: Program aided language models," in Proc. ICML, 2023.
[6] P. Lewis et al., "Retrieval augmented generation for knowledge intensive NLP tasks," in Proc. Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 33, 2020.
[7] A. Dubey et al., "The Llama 3 herd of models," arXiv:2407.21783, 2024.
[8] Gemma Team, "Gemma 2: Improving open language models at a practical size," arXiv:2408.00118, 2024.
[9] R. Smith, "An overview of the Tesseract OCR engine," in Proc. Int. Conf. Doc. Anal. Recognit. (ICDAR), 2007, pp. 629–633.
[10] A. Vaswani et al., "Attention is all you need," in Proc. NeurIPS, 2017.
[11] Y. Xu, M. Li, L. Cui, S. Huang, F. Wei, and M. Zhou, "LayoutLM: Pretraining of text and layout for document image understanding," in Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2020, pp. 1192–1200.
[12] G. Kim et al., "OCRfree document understanding transformer," in Proc. Eur. Conf. Comput. Vis. (ECCV), 2022.
[13] U.V. Marti and H. Bunke, "The IAMdatabase: An English sentence database for offline handwriting recognition," Int. J. Doc. Anal. Recognit., vol. 5, no. 1, pp. 39–46, 2002.
[14] A. Kunchukuttan et al., "AI4BharatIndicNLP corpus: Monolingual corpora and word embeddings for Indic languages," arXiv:2005.00085, 2020.
[15] S. Doddapaneni et al., "Towards leaving no Indic language behind: Building monolingual corpora, benchmark and models for Indic languages," in Proc. Annu. Meeting Assoc. Comput. Linguist. (ACL), 2023.
[16] P. Joshi, S. Santy, A. Budhiraja, K. Bali, and M. Choudhury, "The state and fate of linguistic diversity and inclusion in the NLP world," in Proc. ACL, 2020, pp. 6282–6293.
[17] S. Sitaram, K. R. Chandu, S. K. Rallabandi, and A. W. Black, "A survey of codeswitched speech and language processing," arXiv:1904.00784, 2019.
[18] S. Khanuja et al., "GLUECoS: An evaluation benchmark for codeswitched NLP," in Proc. ACL, 2020, pp. 3575–3585.
[19] R. Bommasani et al., "On the opportunities and risks of foundation models," arXiv:2108.07258, 2021.
[20] S. Yao et al., "ReAct: Synergizing reasoning and acting in language models," in Proc. Int. Conf. Learn. Represent. (ICLR), 2023.
[21] T. Schick et al., "Tool former: Language models can teach themselves to use tools," in Proc. NeurIPS, 2023.
[22] W. McKinney, "Data structures for statistical computing in Python," in Proc. 9th Python in Science Conf., 2010, pp. 56–61.
[23] T. Yu et al., "Spider: A largescale human labeled dataset for complex and cross domain semantic parsing and texttoSQL task," in Proc. Conf. Empirical Methods Nat. Lang. Process. (EMNLP), 2018, pp. 3911–3921.

[24] J. Li et al., "Can LLM already serve as a database interface? A big bench for largescale database grounded texttoSQLs," in Proc. NeurIPS Datasets and Benchmarks Track, 2023.

[25] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "QLoRA: Efficient finetuning of quantized LLMs," in Proc. NeurIPS, 2023.

[26] Z. Ji et al., "Survey of hallucination in natural language generation," ACM Comput. Surv., vol. 55, no. 12, art. 248, 2023.

[27] S. J. Taylor and B. Letham, "Forecasting at scale," Am. Stat., vol. 72, no. 1, pp. 37–45, 2018, doi: 10.1080/00031305.2017.1380080.

[28] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "The M4 competition: 100,000 time series and 61 forecasting methods," Int. J. Forecast., vol. 36, no. 1, pp. 54–74, 2020.

[29] B. Lim and S. Zohren, "Timeseries forecasting with deep learning: A survey," Phil. Trans. R. Soc. A, vol. 379, no. 2194, 2021.

[30] R. J. Hyndman and G. Athanasopoulos, Forecasting: Principles and Practice, 3rd ed. Melbourne, Australia: OTexts, 2021.

[31] F. T. Liu, K. M. Ting, and Z.H. Zhou, "Isolation Forest," in Proc. IEEE Int. Conf. Data Mining (ICDM), 2008, pp. 413–422.

[32] F. Pedregosa et al., "Scikitlearn: Machine learning in Python," J. Mach. Learn. Res., vol. 12, pp. 2825–2830, 2011.

[33] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," ACM Comput. Surv., vol. 41, no. 3, art. 15, 2009.

[34] Y. Gao et al., "Retrieval augmented generation for large language models: A survey," arXiv:2312.10997, 2023.

[35] R. Akkiraju et al., "FACTS about building retrieval augmented generation based chatbots," arXiv:2407.07858, 2024.

[36] P. Chhikara et al., "Mem0: Building production ready AI agents with scalable longterm memory," arXiv:2504.19413, 2025.

[37] S. Schelter, D. Lange, P. Schmidt, M. Celikel, F. Biessmann, and A. Grafberger, "Automating largescale data quality verification," Proc. VLDB Endow., vol. 11, no. 12, pp. 1781–1794, 2018.

[38] M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why should I trust you?': Explaining the predictions of any classifier," in Proc. ACM SIGKDD, 2016, pp. 1135–1144.

[39] S. M. Lundberg and S.I. Lee, "A unified approach to interpreting model predictions," in Proc. NeurIPS, 2017.