

COMP-SCI 5588

Data Science Capstone

Professor: Dr Yugyung Lee

Term name: Bug Killers

Feb 05, 2025

Github Link: <https://github.com/nanxuanhui/DSCapstone.git>

Presentation Link:

<https://drive.google.com/file/d/1vxbZOIQn8D5JdoHxGMYjN1K0V6VlrSLB/view?usp=sharing>

Content

| | |
|--|-----------|
| <i>Team Information</i> | <i>2</i> |
| <i>Significance & Motivation.....</i> | <i>3</i> |
| <i>Objective & Innovation</i> | <i>4</i> |
| <i>Related Work.....</i> | <i>5</i> |
| <i>Challenges</i> | <i>7</i> |
| <i>Potential Dataset.....</i> | <i>7</i> |
| <i>Potential Approach & Technologies.....</i> | <i>8</i> |
| <i>Expected Deliverables</i> | <i>9</i> |
| <i>Model Performance and Evaluation Metrics.....</i> | <i>9</i> |
| <i>AI Model Fusion</i> | <i>10</i> |
| <i>Status of the Current Project</i> | <i>10</i> |
| <i>Timeline & Milestones.....</i> | <i>11</i> |
| <i>Reference</i> | <i>11</i> |

Team Information

1. Team Name

Bug Killer

2. Team Members

| Name | Role |
|------------------------------|----------------------|
| Saniya Pandita | UI Developer |
| Jayadithya Nalajala | Full Stack Developer |
| Sai Jahn timer Devabhakthuni | ML and Data Engineer |
| Hui Jin | AI Developer |

3. Individual Responsibilities

| Name | Responsibility |
|------------------------------|---|
| Saniya Pandita | Wireframing and Prototyping, Visual Design |
| Jayadithya Nalajala | Backend API Calls, AI Model Integration |
| Sai Jahn timer Devabhakthuni | Data Preprocessing & Feature Engineering Model Training & Optimization |
| Hui Jin | YOLOv5 Implementation, OpenPose Integration, Pipeline Construction |

4. Skill sets

| Name | Skill |
|---------------------|---------------------------------|
| Saniya Pandita | React.js, HTML, CSS, JavaScript |
| Jayadithya Nalajala | Python, Node.js, REST API's |

| | |
|-----------------------------|---|
| Sai Jahn timer Devabhaktuni | Python, HTML, pandas, numpy, matplotlib lib, Mongo DB, YOLOv5 |
| Hui Jin | YOLOv5, OpenPose, Python, MLX |

5. Personal Contribution

| Name | Contribute |
|-----------------------------|---|
| Saniya Pandita | Lead UI design, optimize React.js interface, and improve user interaction experience. |
| Jayadithya Nalajala | Develop and integrate backend APIs, using Flask and FastAPI to connect frontends with AI models. |
| Sai Jahn timer Devabhaktuni | Process the dataset to ensure data cleaning, structure optimization, and optimize multimodal learning data. |
| Hui Jin | Lead the integration of YOLOv5 + OpenPose to improve the accuracy of fall detection and optimize inference performance. |

Significance & Motivation

1. Importance

Elderly people, especially those living alone, face significant health risks due to falls and mental health concerns. Falls are a leading cause of injury-related hospitalizations, while social isolation and cognitive decline are growing concerns among the aging population. An AI-driven system that detects falls and provides mental health companionship can significantly improve elderly care.

2. Problem

Fall Detection: Real-time identification and alerting of falls using AI vision models.

Mental Health Companion: AI-driven chatbot to provide emotional support and cognitive engagement.

3. Benefits

Elderly individuals living alone

Family members & caregivers

Healthcare providers

Assisted living facilities

Objective & Innovation

1. Main goal

Develop a real-time fall detection system using AI vision models.

Integrate an AI-powered mental health chatbot to provide cognitive support.

Develop an intuitive web UI for caregivers and users to monitor and interact with the system.

Ensure seamless API integration between the AI models, chatbot, and database

2. Difference from existing solutions

This system combines AI-powered fall detection and a mental health chatbot to provide both safety and emotional support.

Unlike solutions that rely on wearable devices, our YOLOv5-based vision model can achieve high-precision, real-time detection without the need for additional hardware.

The chatbots are specially trained to enhance user interaction experience and provide mental health support.

Our system is easier to deploy and scale than traditional solutions.

3. New

Combination of vision-based fall detection and natural language processing (NLP) chatbot in one system.

Use of YOLOv5 for real-time object detection in fall identification.

Chatbot powered by Transformer-based models trained for elderly support.

Cloud-based MongoDB for efficient data storage and retrieval.

4. Project Goals

• Core Goals

Focus on computer vision-based fall detection and mental health support with AI conversational assistants.

• Target Evolution

Based on the original YOLOv5 model, explore the performance comparison of YOLOv7, YOLOv10, and YOLOv11.

Optimize OpenPose integration to improve accuracy under different lighting and angles.

Reduce computing costs and improve API call efficiency through lightweight fine-tuning methods such as LoRA.

5. Hands-on Learning Contributions

RAG (Retrieval-Augmented Generation) is used to enhance the chatbot so that it can answer questions based on an external knowledge base.

Fine-tune the LoRA (Low-Rank Adaptation) technology to improve the adaptability of the dialogue model under low-resource conditions.

6. Technical Enhancements

- Expand the dataset and synthesize data through GAN (generative adversarial network) to improve the generalization ability of the model.
- Enhance multimodal understanding (LMMs, Large Multimodal Models), combining speech and text analysis.

7. Planned Improvements

- Add ARKit to increase recognition accuracy.
- Improve the quality of conversations in a specific domain (mental health of the elderly) through LoRA fine-tuning.
- Study how to run YOLO efficiently on Apple Vision Pro.

Related Work

1. Key findings

The fall detection system using OpenPose and YOLO achieved 100% precision, recall, F1 score, and mAP, demonstrating exceptional performance in detecting falls among the elderly. The integration of OpenPose improved accuracy in low-light conditions, achieving a fall detection accuracy of 99.75%.[1]

The EBER chatbot significantly improved user satisfaction among elderly participants, with 80% rating the system a score of 4 out of 5, indicating its effectiveness as an "intelligent radio" for entertainment and companionship. The system demonstrated the ability to adapt to users' moods through Sentiment Analysis (SA), which enhanced the conversational experience and engagement. The chatbot's design, which intersperses news reading with light dialogues, was well-received, as it aligned with the traditional media consumption habits of the elderly, thus reducing feelings of loneliness.[2]

The study presents a novel conversational system that effectively monitors cognitive impairment in elderly individuals through an entertainment chatbot. The system achieved a detection accuracy of cognitive impairment close to 90% using Machine Learning algorithms, demonstrating strong potential for user-friendly therapeutic monitoring. Additionally, the results indicated that users without cognitive impairment performed significantly better than those with mild or severe impairments, with a similarity metric ranging from 0.03 for stressed users to 0.36 for relaxed users.[3]

2. Methodologies

The study utilized a dataset of 430 images, divided into categories for training and testing, with a focus on comparing systems with and without OpenPose. YOLO was employed for real-time object detection, processing images through convolution and max pooling to generate bounding boxes.[1]

The EBER chatbot employs a combination of technologies, including Artificial Intelligence Markup Language (AIML), Natural Language Generation (NLG), and Sentiment Analysis (SA), to create coherent and contextually relevant dialogues. Experimental tests were conducted with 31 elderly users, assessing their interactions with the chatbot and measuring satisfaction, confusion, and stress levels through a structured feedback process. The system utilized facial recognition and voice commands for user interaction, making it accessible and user-friendly for the elderly population.[2]

The researchers employed a combination of Natural Language Processing (NLP) techniques and Machine Learning algorithms to create a chatbot that engages users with news content while assessing their cognitive capabilities. The system generates questions based on news items and evaluates user responses by comparing them to a gold standard derived from the same content. Field tests were conducted with 30

elderly participants under the supervision of gerontologists, focusing on the effects of stress and concentration on cognitive assessment.[3]

3. Gaps

The research indicates a need for more diverse datasets and the exploration of augmentation techniques, such as Generative Adversarial Networks (GAN). Future studies could benefit from using high-spec hardware to enhance training speed and model performance.[1]

Despite the positive feedback, the chatbot's ability to detect when a user has finished speaking needs improvement, as some users reported confusion due to interruptions. The current implementation does not fully mimic human-like empathy, which could further enhance user engagement and satisfaction. There is a need for further research to explore the long-term effects of using the chatbot on users' cognitive abilities and abstraction skills, particularly among those with varying levels of technological proficiency.[2]

Despite the advancements in AI and NLP, existing intelligent systems for cognitive assessment still rely heavily on predefined manual tests, which can be time-consuming and may induce stress in users, known as the "white-coat effect". The study highlights that while there are intelligent systems in telecare, they often lack the capability for autonomous communication with the elderly using natural language, indicating a significant gap in empathetic interaction. Furthermore, the research suggests that more work is needed to enhance the system's empathetic capabilities through user feedback, which is currently limited.[3]

4. Literature Review and Novelty

- **How to build**

- Combine existing research results on computer vision fall detection and natural language processing (NLP) mental health chat.
- Combine the progress of Transformer architecture (such as RAG, LoRA) to improve chatbots and enhance interaction capabilities and user experience.

- **Challenges**

- Fall detection requires low latency and high accuracy to ensure that there are no missed or false detections.
- The elderly's acceptance and usage habits of AI systems affect the effectiveness of the system.

- **Novel methods or applications**

- Adopt LoRA (Low-Rank Adaptation) technology to optimize chatbot performance in resource-constrained environments.
- Combined with multimodal learning (LMMs), the system can simultaneously understand and analyze multiple data inputs such as voice, text, and video.
- Use edge computing and serverless architecture to reduce computing costs and improve system scalability.

Challenges

1. Technical or conceptual challenges

- Real-time processing of fall detection
- Ensuring chatbot adaptability for elderly users
- Data privacy and security
- Handling false positives in fall detection

2. Plan to overcome

- Real-time processing of fall detection
- Ensuring chatbot adaptability for elderly users
- Data privacy and security
- Handling false positives in fall detection

Potential Dataset

1. Data sources, size, and format

The dataset is from kaggle and is based on a survey designed to study depression and mental health across various demographics. It includes information from a wide range of sources, including teenagers from Bangladesh, college students, housewives, professionals from businesses and corporations, and other people.

The size of the dataset is 62.39 KB.

The dataset is in csv format.

<https://www.kaggle.com/datasets/shashwatwork/depression-and-mental-health-data-analysis/data>

Data Sources: The dataset consists of 8,713 fall images collected from the internet, covering various environments and fall types to improve the model's generalization.

Size: A total of 8,713 images were collected and annotated for training and validation. The dataset was split into an 80:20 ratio for training and validation.

Format: The dataset was labeled using the YOLO format, with annotations created using Labellmg. The bounding boxes were manually drawn around fall-related instances and saved as .txt files corresponding to the images.

2. Data preprocessing

- Handle any missing values by removing or filling them. Ensure that each column has the correct data type, such as numeric values.
- Use encoding techniques to convert categorical data, such as gender, to numeric format. If necessary, scale numeric features to standardize their values. Remove any irrelevant or unnecessary columns that do not contribute to the analysis. Check and eliminate duplicate rows to maintain unique data.
- Further remove invalid or low-quality data to reduce the impact of noise.
- Use feature selection techniques, such as SHAP analysis, to optimize input variables and reduce redundant data.
- Split the dataset into training and test sets for machine learning.

3. Challenges and Resolutions

- Since data formats from different sources are inconsistent, an automatic data format conversion tool is used for standardization.
- Use data augmentation (such as image flipping and random cropping) to improve the generalization ability of the model.

Potential Approach & Technologies

1. Technologies, frameworks, and models

Machine Learning & AI: YOLOv5 for fall detection, Transformer-Models, Tensorflow, Pytorch.

Frontend Development: React.js for an intuitive user interface.

Backend Development: Python with Flask/FastAPI for REST API development, Node.js for react server.

Database Management: MongoDB for efficient data storage and retrieval.

2. Programming languages, tools, and platforms

Languages: Python, JavaScript (React.js), CSS.

Tools: GitHub, Docker (for containerization).

Platforms: Web application.

3. APIs and external services

AI APIs: DeepSeek API, OpenAI API.

Twilio API: SMS alerts API.

4. Technical Architecture

• Architecture Evolution

- The initial architectural design evolved into a modular microservice architecture, which improved scalability and maintainability.
- The integration of LoRA and RAG enables the AI dialogue system to perform dynamic knowledge retrieval.

• Data Flow and Processing

- Use streaming data pipelines to improve data processing efficiency.
- Optimize API endpoints through FastAPI to reduce data transmission latency.

• Advances in Deep Learning and AI Models

- Improve YOLOv5 detection accuracy using enhanced datasets and ARKit technology.
- Incorporate Transformer architecture into mental health chatbots to improve natural language processing capabilities.

• Component Integration

- RAG (Retrieval Enhanced Generation) is used to provide more accurate information query.
- LoRA technology optimizes model fine-tuning and reduces computing power requirements.

• Deployment and UI development progress

- Use Flask+FastAPI to complete the backend API construction.
- Use Streamlit for rapid prototyping to achieve seamless interaction between the frontend and the backend.

- Add data visualization function to the React.js frontend to optimize the user experience.

Expected Deliverables

1. Final output

Combine YOLOv5 and ARKit to provide high-precision, real-time fall detection and reduce false positives and missed detections.

Use Transformer models and sentiment analysis to provide emotional support and mental health advice for the elderly.

Provide a user-friendly monitoring interface so that caregivers and users can view data in real time.

Includes data storage and calling solutions for Python (Flask/FastAPI) and MongoDB.

Record the complete development process, technical architecture, and model training process.

2. Key features or functionalities

AI-Powered Fall Detection: Use YOLOv5+ARKit for real-time, device-free monitoring to improve detection accuracy.

Mental Health Companion Chatbot: Based on NLP and RAG technology, the chat quality is optimized and the naturalness of human-computer interaction is improved.

Real-Time Alerts & Notifications: Twilio integration for immediate caregiver intervention.

Intuitive Web Dashboard: React.js UI for monitoring and user interaction. Model Performance and Evaluation Metrics.

3. Contribution of practical learning to final realization

- Through Transformer and LoRA fine-tuning, the chatbot's sentiment analysis and context understanding capabilities were optimized.
- Combining FastAPI and Streamlit improved the API call efficiency and the interactive experience on the Web side.

4. Future Development and Major Milestones

- Further improve the detection performance of YOLOv5+ARKit and optimize the knowledge retrieval mechanism of RAG.
- Optimize the FastAPI interface, increase the system call speed, and achieve scalable deployment in the cloud.
- Optimize deployment on Apple Vision Pro to achieve low power consumption and high performance computing.

Model Performance and Evaluation Metrics

1. Evaluation Metrics

- The YOLOv5 fall detection model achieved a mAP (mean average precision) of 90%.
- The mental health chatbot's BLEU score improved to 0.78 and ROUGE-L to 0.82.

2. Challenges and Optimization

- Expand fall detection samples through data augmentation (such as GAN generated data).
- Enhance the context understanding ability of the dialogue model and reduce the generation of irrelevant responses.

3. Future optimization direction

- Integrate RAG (Retrieval Augmentation Generation) to enable chatbots to provide more accurate mental health advice.
- Further optimize LoRA fine-tuning technology to improve model performance in low computing resource environments.

AI Model Fusion

1. Fusion Strategy

- Combine YOLOv5 with the detection features of ARKit to improve the accuracy of fall detection.

2. Models and challenges for successful integration

- The combination of YOLOv5 and ARKit improves the accuracy and real-time performance of fall detection.
- Computing resource limitations limit real-time processing capabilities.
- It is necessary to further optimize the integration of YOLOv5 and ARKit to improve the interaction efficiency between different AI modules.

3. Fusion improves accuracy and prediction

- Combining YOLOv5 and ARKit can improve the robustness of fall detection and reduce misjudgments caused by environmental changes.

Status of the Current Project

1. Completed tasks

- The training of the fall detection model based on YOLOv5 has been completed, and the current confidence level has reached 85%.
- In order to solve the problem of false detection, ARKit is introduced to improve the detection accuracy.

2. Challenges

- During training, modifying the configuration sometimes leads to a decrease in confidence, and a more stable optimization strategy is needed.
- The problem of false detection still exists, and the data enhancement strategy needs to be further adjusted.

3. Work in Progress

- Try different hyperparameter tuning methods to improve model performance.
- The training of the Speech Emotion dataset is used to optimize the emotion recognition ability of AI assistants and improve the naturalness of human-computer interaction.

4. Next steps

- Adopt RAG (retrieval-augmented generation) technology to improve the dynamic knowledge calling capability of the dialogue system.
- Combined with LoRA (low-rank adaptation) fine-tuning technology to reduce computing costs and improve performance.

Timeline & Milestones

1. Week-by-week breakdown of tasks

| Week | Task |
|---------|---|
| Week 3 | Preprocess data for the selected idea. |
| Week 4 | Train and refine the chosen AI model. |
| Week 5 | Develop core functionalities (fall detection or chatbot). |
| Week 6 | Build backend API integration. |
| Week 7 | Develop React.js frontend. |
| Week 8 | Integrate AI model with the system. |
| Week 9 | Implement real-time alerts (if applicable). |
| Week 10 | Conduct system testing and improvements. |
| Week 11 | Optimize performance and finalize features. |
| Week 12 | Scientific Paper Writing |

Reference

[1] Hardiyanto, Ridho Adha, et al. "Empowering Elderly Care: Innovative Fall Detection with OpenPose and YOLO." 2023 6th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI). IEEE, 2023.

[2] García-Méndez, Silvia, et al. "Entertainment chatbot for the digital inclusion of elderly people without abstraction capabilities." IEEE Access 9 (2021): 75878-75891.

[3] de Arriba-Pérez, Francisco, et al. "Automatic detection of cognitive impairment in elderly people using an entertainment chatbot with Natural Language Processing capabilities." Journal of ambient intelligence and humanized computing 14.12 (2023): 16283-16298.