Lost and Found Using Deep Learning

A Synopsis Submitted

in Partial Fulfillment of the Requirements

for the Degree of

BACHELOR OF TECHNOLOGY

in

Department of Artificial Intelligence & Artificial Intelligence & Machine Learning

by

Abhijeet Warale Roll No 01
Arpeet Chandane Roll No:16
Abhishek Kasture Roll No:04
Abhishek Kadam Roll No:03

GHraisoni COLLEGE
Engineering and Management
Pune

G H Raisoni College of Engineering and Management

Wagholi, Pune

Department of Artificial Intelligence and Artificial Intelligence & Machine Learning

(An Empowered Autonomous Institute (NAAC A+ Grade) Affiliated to SPPU, Pune)

June, 2024

Synopsis

1. Introduction

In today's fast-paced world, losing personal items is a common issue, leading to stress and inconvenience. Conventional lost-and-found systems rely on manual reporting, making the process slow and ineffective. To address this problem, we propose an AI-powered Lost and Found System that leverages state-of-the-art deep learning models to automatically match lost items with found items. This system uses a combination of image captioning and natural language processing (NLP) techniques, providing an efficient solution that works at scale.
The system is built upon the BLIP (Bootstrapping Language-Image Pre-training) model, a fine-tuned image-captioning neural network, and BERT (Bidirectional Encoder Representations from Transformers), a robust language model for comparing textual descriptions. By integrating these models, the system can analyze images of found items, generate descriptive captions, and match these captions against user-submitted descriptions of lost items.
The project's uniqueness lies in its use of deep learning to automate the entire process—from identifying found items to notifying users when a match is found. This application not only improves the efficiency of lost-and-found services but also provides a user-friendly interface for seamless interaction.

2. Objectives

The objectives of the Lost and Found items app could include the following:

1. **Automated Image Captioning:**

- Leverage the BLIP model to generate accurate descriptions of images representing found items. These captions will include detailed object characteristics to maximize matching accuracy.

2. **Textual Matching Using NLP:**

- Employ BERT to process the descriptions submitted by users looking for lost items. The system compares these descriptions with the generated captions to find the best matches based on a cosine similarity index.

3. **Seamless User Experience:**

- Develop an intuitive and easy-to-navigate user interface for reporting lost items and uploading images of found items, making the system accessible to users of all technical backgrounds.

4. **Real-Time Matching and Notification:**

- Provide real-time alerts and notifications to users when a match between a lost and found item is detected. This ensures timely communication between users and the system.

3.Literature Review:

The application of deep learning in lost-and-found systems has grown significantly with advancements in computer vision and NLP. Traditional methods of manually searching for lost items in databases are being replaced by intelligent systems that automatically process and match visual and textual data.

i. **Convolutional Neural Networks (CNNs)** are widely used for object detection and image recognition, making them ideal for identifying objects in images. CNNs can be trained to recognize various categories of objects (e.g., phones, wallets, keys) that are commonly lost.

ii. **Long Short-Term Memory (LSTM) networks** are employed alongside CNNs to capture the temporal dependencies in data, such as the time and location of an item's last known position.

iii. **Image captioning systems**, such as **BLIP**, generate natural language descriptions from visual data, translating the content of an image into text. By training on large-scale datasets such as **COCO**, BLIP models can describe a wide variety of objects and scenarios accurately.

iv. **BERT**, a powerful NLP model, is used for comparing text descriptions. It excels at understanding the **semantic meaning** of text, enabling the system to match user-provided descriptions with captions generated from images.

Research shows that combining **CNNs** with **LSTM** models significantly enhances the system's ability to identify lost items in complex environments, such as crowded public spaces, airports, or shopping malls. However, challenges such as **varying lighting conditions** and **similar-looking objects** persist, requiring further research to improve accuracy.

Sr no.	Paper Title	Year	Details of Publication	Findings
1	Deep Learning Approaches on Image Captioning: A Review	2023	*arXiv:2201.12944v5 [cs.CV]*	This paper provides an extensive review of deep learning techniques for image captioning, focusing on methods such as **CNN-LSTM** and attention-based approaches.
2	Ensemble Learning on Deep Neural Networks for Image Caption Generation	2022	*14th IEEE International Conference on Semantic Computing (ICSC)*	Demonstrates that **ensemble learning** techniques, when applied to deep neural networks, significantly improve the performance of **image captioning** models by combining multiple neural network outputs.
3	A Survey on Deep Neural Network-Based Image Captioning.	2019	*The Visual Computer* 35, no. 3 (2019): 445–470.	Outlines recent trends in **deep learning-based image captioning**, highlighting how the combination of **CNNs** and **RNNs (including LSTMs)** has improved caption generation tasks by learning complex visual and textual representations.

4.Methodology:

The project's methodology focuses on developing a system capable of generating captions for found items, comparing those captions with user-provided descriptions of lost items, and notifying users when a match is found. The steps involved are:

4.1 Data Collection and Preparation –

I. Data Sources: The system's dataset is curated using images of common daily items extracted from the COCO dataset and Google Image Scraping. Items include frequently lost objects such as keys, wallets, mobile phones, sunglasses, and bags.

II. Data Preprocessing: The images are preprocessed by resizing and normalizing them to fit the input dimensions required by the BLIP model. Captions for the images are saved in COCO Captions Format, ensuring uniformity for the model training process.

4.2 Model Development –

I. BLIP Model for Image Captioning: The BLIP model is fine-tuned on the curated dataset to generate accurate and detailed captions for the found items. BLIP's architecture combines Convolutional Neural Networks (CNNs) for feature extraction and Transformer models for sentence generation, making it a powerful tool for image-to-text tasks.

II. BERT Model for Textual Comparison: BERT is employed to analyze the descriptions of lost items submitted by users. BERT's deep understanding of contextual language allows it to create robust sentence embeddings, which are compared with the generated captions using cosine similarity.

III. Matching Algorithm: A matching algorithm compares the sentence embeddings of user-submitted descriptions and BLIP-generated captions. If the cosine similarity score exceeds a defined threshold (e.g., 0.6), the system considers it a potential match and notifies the user.

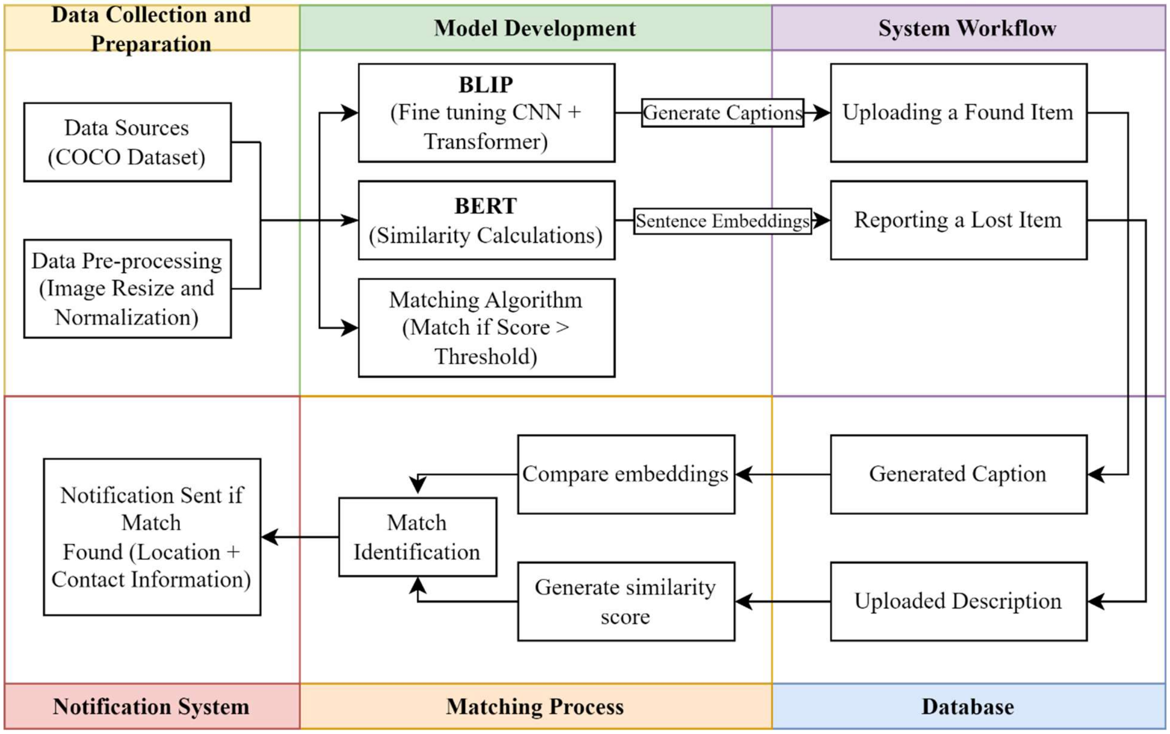
4.3 System Workflow –

I. Uploading a Found Item: Users upload images of found items via the app. The BLIP model generates captions, which are stored in a SQLite database along with the item's image and metadata (e.g., location, contact details of the finder).

II. Reporting a Lost Item: Users who have lost an item provide a detailed description, which the system processes using BERT to generate a sentence embedding.

III. Matching Lost and Found Items: The system continuously scans the database for matching captions using the cosine similarity metric between user descriptions and found item captions. When a match is found, the user is notified via email, providing them with the found item's location and the contact details of the person who found it.

Flowchart for Methodology:



5.Project Plan and Timeline:

Month 1: Planning and Data Preparation

- **Objectives**: Define the project, gather data, and prepare it for model training.
    - **Project Planning**: Define scope, objectives, and gather requirements from stakeholders.
    - **Data Collection**: Collect images of lost and found items and metadata (time, location).
    - **Data Annotation**: Label and categorize images; enhance dataset with data augmentation.
    - **Data Preprocessing**: Resize, normalize images, and clean metadata for model input.

Month 2: Model Development:

- **Objectives**: Develop and train the CNN and LSTM models.
- **CNN Model Development**:
    - Design and train a CNN for image recognition and feature extraction.
    - Fine-tune hyperparameters for optimal performance.
    - Start training the CNN with preprocessed image data.
    - Monitor training performance and adjust hyperparameters as needed.

- **LSTM Model Development:**
  - Design the LSTM architecture to handle sequence data (e.g., time and location).
  - Implement the model to process and analyze temporal patterns in lost and found data.
  - Train the LSTM model using sequence data related to item reports and matches.

- **LSTM Model Fine-Tuning:**
  - Evaluate the LSTM model's performance and adjust hyperparameters.
  - Integrate LSTM outputs with CNN outputs for improved matching accuracy.
  - Prepare models for integration with the application backend.

Month 3: System Development:

- **Backend Development:**
  - Develop the backend infrastructure, including server setup, database design, and API development.
  - Integrate the trained CNN and LSTM models for real-time processing and inference.
  - Implement data storage solutions for images, metadata, and model results.
- **Backend Integration:**
  - Test the integration of models with backend services.
  - Ensure smooth data flow between the frontend, backend, and models.
  - Set up security measures to protect user data.
- **Frontend Development:**
  - Design and develop the user interface, focusing on usability and aesthetics.
  - Implement features for users to report lost and found items, upload images, and view matches.
  - Develop notifications and search functionalities.

Month 4: Testing and Integration:

- **Model Testing and Validation:**
  - Conduct extensive testing of the CNN and LSTM models using separate test datasets.
  - Evaluate model accuracy, precision, and recall.
  - Refine models based on test results and re-train if necessary.

- **System Testing:**
  - Perform end-to-end testing of the application, including user interactions, data processing, and notifications.
  - Test system performance under various conditions (e.g., load testing, stress testing).
  - Identify and fix bugs or performance issues.

- **User Testing:**
  - Launch a beta version of the app to a select group of users.
  - Collect feedback on usability, functionality, and overall user experience.
  - Analyze feedback and prioritize changes or improvements.

- **Refinements:**
  - Implement changes based on user feedback.
  - Finalize the app's features and address any remaining issues.
  - Prepare documentation and training materials for deployment.

Month 5: Deployment and Maintenance:

- **Final Improvements:**
  - Make final adjustments to the app based on testing and feedback.
  - Ensure all features are functioning correctly and the app is ready for launch.

- **Deployment:**
  - Deploy the app to a cloud platform or app store.
  - Set up continuous integration/continuous deployment (CI/CD) pipelines for future updates.
  - Ensure proper configuration of deployment environments.

- **Monitoring and Maintenance Setup:**
  - Establish monitoring tools to track app performance, user activity, and model accuracy.
  - Set up procedures for ongoing maintenance, including bug fixes and performance tuning.

- **Launch and Support:**
  - Provide user support and address any issues that arise post-launch.
  - Collect initial user feedback and plan for future updates and improvements.

6.Expected Outcomes:

The successful completion of this project is expected to result in:

I. **Automated Caption Generation**: The system will generate detailed descriptions of found items using the **BLIP** model.
II. **Efficient Database Management**: The captions and images of found items will be stored in a centralized database for easy access and comparison.
III. **High Matching Accuracy**: The integration of **BERT** and **cosine similarity** will ensure high accuracy in matching lost and found items.
IV. **Real-Time Notifications**: Users will receive immediate alerts when a potential match for their lost item is found.
V. **User Satisfaction**: By reducing the time and effort required to find lost items, the system will significantly enhance user satisfaction.

7.References:

- P. Choudhary, A. Choudhary, A. Singh, “Find the Lost Items via Mobile App”, 2021 2nd International Conference on Intelligent Engineering and Management (ICIEM), India, 2021, pp. 1-6

- Sakib, S.Mukta, Md.Hossain, “Automated Image Captioning System”, Independent University, Bangladesh, April 2024, DDI:10.13140/RG.2.224966

- G. Geeta, T. Kirtigadevi, G. Godwin Ponsam, T. Karthik and M. Safa, “Image Captioning Using Deep Convolutional Neural Networks (CNNs), Journal of Physics: Conferrence Series, Vol. 1712, pp. 012015, Dec. 2020

- Weller, Mira, et al. "Lost and Found: Stopping Bluetooth Finders from Leaking Private Information." arXiv preprint arXiv:2005.08208 (2020).

- Price, T., 2008. Lost item notification and recovery system. U.S. Patent Application 11/542,447. [11] Bhardwaj, G., Singh, S.V. and Kumar, V., 2020, January. An Empirical Study of Artificial Intelligence and its Impact on Human Resource Functions. In 2020 International Conference on Computation, Automation and Knowledge Management (ICCAKM) (pp. 47-51). IEEE.

- Ortiz, S.A. and Ortiz, L.M., Open Invention Network LLC, 2018. Determining the last location of lost and stolen portable electronic devices when wireless communications access to the lost or stolen devices is lost or intermittent. U.S. Patent 9,860,703.

- F. Muhammad Shah, M. Humaira, J. Jim, A. S. Ami, and S. Paul, ”Bornon: Bengali Image Captioning with Transformer-based Deep learning approach,” arXiv (Cornell University), Sep. 2021. doi: 10.48550/arxiv.2109.05218.

- S. Liu, L. Bai, Y. Hu, and H. Wang, ”Image Captioning Based on Deep Neural Networks,” MATEC Web of Conferences, vol. 232, pp. 01052, 2018. doi: 10.1051/matecconf/201823201052.

- P. Anderson, B. Fernando, M. Johnson, and S. Gould. SPICE: Semantic propositional image caption evaluation. In European Conference on Computer Vision, pages 382–398. Springer, 2016.

- Hasan, M., Alfaz, N., Alam, M., Rahman, A., Shakhawat, H. & Rahman, S. Detection of Parkinson’s Disease from T2-Weighted Magnetic Resonance Imaging Scans Using EfficientNet-V2. 2023 26th International Conference On Computer And Information Technology (ICCIT). pp. 1-6 (2023)

Project Guide	**HOD**
Prof. Swati Barik	***Dr. Rachana Sable***