MULTIMODAL DATA MANAGEMENT IN OPEN-WORLD ENVIRONMENT

by

KMA Solaiman

A Dissertation

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Doctor of Philosophy

Department of Computer Science

West Lafayette, Indiana

August 2023

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Bharat Bhargava, Chair

Department of Computer Science

Dr. Vaneet Aggarwal

School of Industrial Engineering

Dr. Chunyi Peng

Department of Computer Science

Dr. Jianguo Wang

Department of Computer Science

Dr. Xavier Tricoche

Department of Computer Science

Approved by:

Dr. Kihong Park

To my father Md Ali, my mother Panna, my niece Saffana, my family members who sacrificed many precious years with me, and my dear friends who has helped me survive the toughest years of my life.

ACKNOWLEDGMENTS

First of all, I would like to express my deep sense of appreciation to my advisor Prof. Bharat Bhargava for his continuous support and guidance throughout this journey. I would like to thank Prof. Michael Stonebraker for imparting his knowledge and expertise and for being a mentor throughout the REALM project. The completion of this study would not have been possible without your valuable contributions. I would also like to thank my esteemed committee members for their continuous support and advice.

This research is supported by Northrop Grumman Mission Systems University Research Program. Thanks is also due to everyone with whom we collaborated on this research, Prof. Michael Cafarella from University of Michigan, Jim MacDonald and Jason Kobes from NGC, Sergeant Troy Greene, and all the team members on the REALM project from Purdue University and MIT.

This research is supported, in part, by the Defense Advanced Research Projects Agency (DARPA) and the Air Force Research Laboratory (AFRL) under the contract number W911NF2020003. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA, AFRL, or the U.S. Government.

Thanks is also due to everyone with whom we collaborated on this research, Prof. Mayank Kejriwal and his team at USC, Prof. Vaneet Aggarwal, all the team members on the SAIL-ON project and the Novelty Working Groups for all the discussions that contributed to developing this research, specially Terry Boult, Josh Alspector, Katarina Doctor, and Eric Kildeleck. We also gained a great deal of knowledge at the AAAI Symposium led by Mayank: Designing Artificial Intelligence for Open Worlds as well as the Workshop on Novelties in Open World organized by Prof. Bhargava, in conjunction with International Semantic Intelligence Conference (ISIC-2021).

Most definitely I would not have made it this far without the support of my amazing family. I am grateful to my mom, dad, my sister and sister-in-law, and my niece for their love, prayers and unshaken belief in me despite all the difficulties. A special thanks to my extended family and friends for their genuine support, encouragement, and honest advice

throughout this journey. There were few who helped me survive the most difficult part of my life and kept me moving forward. Words will fall short to express my deep gratitude to everyone who supported me.

Finally, my heartfelt gratitude to God Almighty, “Allah, The Most Gracious and The Most Merciful”, for all his blessings, forgiveness, and for giving me patience and wisdom to complete this long journey.

TABLE OF CONTENTS

LIST OF TABLES . . . 12

LIST OF FIGURES . . . 13

ABBREVIATIONS . . . 15

ABSTRACT . . . 16

1 INTRODUCTION . . . 18

1.1 Background and Motivation . . . 18

1.2 Dissertation Contribution . . . 19

1.2.1 Decision Making Information Extraction from Multimodal Data . . . 20

1.2.1.a Situational Knowledge Query Engine Framework . . . 20

1.2.1.b Novel Use Case for West Lafayette Police Department . . . 21

1.2.1.c Novel Human Attribute Recognition model . . . 21

1.2.2 Data Integration and Relevance Learning . . . 22

1.2.2.a Applying Schema Mapping and Data Fusion to “Missing Person” Problem . . . 23

1.2.2.b Feature-centric Multimodal Information Retrieval with Graph Matching . . . 23

1.2.2.c Weakly Supervised Joint Embedding for Multimodal Information Retrieval . . . 24

1.2.3 Uncertainty Management in Multimodal Information Retrieval . . . 24

1.3 Dissertation Organization . . . 25

2 DATA-DRIVEN DECISION MAKING INFORMATION EXTRACTION FROM MULTIMODAL DATA . . . 26

2.1 Situation Knowledge on Demand (SKOD) . . . 26

2.1.1 Example Application Scenario . . . 27

2.1.2 SKOD System Architecture . . . 28

2.1.2.a Streaming Broker 28
2.1.2.b Multimodal Query Engine 30
Feature Extraction from Image and Video Streams 31
Feature Extraction from Text 31
Data Integration and Knowledge Modeling 32
2.1.2.c Data Retrieval and Missing Information Completion 34
2.1.3 Prototype Implementation 34
2.1.3.a Dataset 34
2.1.3.b Experimental Settings 36
High-level Feature Extraction from Text and Video 36
2.1.3.c Front End and Indexing Layer 38
2.1.4 Related Work 38
2.1.5 Conclusion and Future Work 40
2.2 Surveillance Video Querying With A Human-in-the-Loop (SurvQ) 40
2.2.1 Introduction 40
2.2.1.a Surveillance Video and Detective Use Cases 42
2.2.2 Related Work 44
2.2.3 Architecture 46
2.2.3.a Data Ingestion 46
2.2.3.b Granular-level Property Identification in Videos 46
2.2.3.c Data Retrieval with Incomplete Data and Meeting Information Needs Over Time 48
2.2.4 User Workflow: The Police Detective In The Loop 48
2.2.5 Runtime Performance: Scalability of Data Ingestion and Retrieval . . 50
2.2.6 Conclusion and Future Work 51
2.3 Human Attribute Recognition from Unstructured Text (HART) 52
2.3.1 Problem Definition 52
2.3.2 Methodology 53
2.3.2.a Candidate Sentence Extraction 54
(i) Pattern Matching. 54

(ii) Similarity using Tokens 55
(iii) Classification Model 55
(iv) Stacked Models 56
2.3.2.b Iterative Search for Properties 56
Generalization. 61
2.3.3 Experiments 61
2.3.3.a Dataset Construction. 61
2.3.3.b Settings. 61
2.3.3.c Results 62
2.3.4 Conclusion and Future Works 63
3 APPLYING MACHINE LEARNING AND DATA FUSION TO THE “MISSING PERSON” PROBLEM 74
3.1 Introduction 74
3.2 Related Work 77
3.2.1 Missing Person Search Applications. 77
3.2.2 Person Re-identification (RE-ID). 78
3.2.3 Cross-modal Matching and Correlation Learning. 78
3.3 System Overview 79
3.3.1 Data Ingestion 80
3.3.1.a Data Capture. 80
3.3.1.b Pre-processing of Video Feeds. 81
3.3.1.c Pre-processing for Unstructured Text and Tweets. 81
3.3.1.d Data Storage. 82
3.3.2 Relevance Modeling and Data Fusion (EARS) 82
3.3.2.a Entity-Attribute-Relationship Model with Schema Mapping. 82
3.3.2.b Data Fusion with SQL JOIN. 83
3.3.2.c User Preference Modeling. 83
3.3.3 Data Retrieval 84
3.4 Feature Extraction 84

3.4.1 Color Analysis for Body Details. 86
3.4.2 Re-training YOLO. 87
3.4.3 Human Attributes from Unstructured Text. 87
3.4.4 Semantic Similarity Search by Topic. 88
3.5 Demonstration 89
3.6 Scalability, Universality, and Multi-user. 90
3.7 Conclusion and Future Work 91

4 FEATURE-CENTRIC MULTIMODAL INFORMATION RETRIEVAL (FEMMIR) 96
4.1 Introduction 96
4.1.1 Preliminaries & Problem Definition 103
4.2 Multimodal Information Retrieval 104
4.2.1 Data Ingestion with Graphs 105
4.2.2 Weak Label Generation 109
4.2.3 Similarity Measurement 112
4.2.4 FemmIR algorithm 113
Generalization. 116
4.3 Experiments 117
4.3.1 Dataset 117
4.3.2 Settings 118
4.3.3 Benchmarking for Property Identifiers in Visual Modality 118
4.3.4 Retrieval Performance of FemmIR 119
4.4 Related Works 120
4.4.1 Metric Learning. 120
4.4.2 Weakly Supervised Learning. 120
4.4.3 Semantic Understanding with Encoding Networks. 120
4.4.4 Content-based Data Discovery. 121
4.4.5 Cross-modal Correlation Learning. 121
4.5 Summary and Future Directions 122

5 WEAKLY SUPERVISED JOINT EMBEDDING FOR MULTI-MODAL INFORMATION RETRIEVAL (WESJEM) . . . 126
5.1 Introduction . . . 126
5.2 Related Works . . . 128
5.3 Methodology . . . 129
5.3.1 Problem Formulation and Overview . . . 129
5.3.2 Translation Module . . . 131
5.3.3 Information Network across Data Samples . . . 132
5.3.4 Multi-Task Learning . . . 133
5.3.5 Reasoning Over the Data Information Network . . . 136
Weak Supervised Baseline. . . . 136
Similarity Based Score. . . . 137
5.4 Experiments . . . 137
5.4.1 Negative Sampling . . . 138
5.4.2 Dataset and Experimental Setup . . . 139
5.4.3 Results. . . . 139
5.5 Conclusion and Future Works . . . 140
6 UNCERTAINTY MANAGEMENT IN DATA-DRIVEN APPLICATIONS . . . 142
6.1 Formalization of Novelties in Multimodal Information Retrieval . . . 142
6.1.1 Task Definition . . . 142
6.1.2 Data Shift in Multimodal Data Retrieval Task . . . 142
6.1.3 Novelty Detection in WeSJem . . . 143
6.1.4 Novelty Characterization for Multimodal Information Retrieval . . . 144
6.1.5 Novelty Response in WeSJem . . . 144
7 CONCLUSION . . . 146
7.1 Future Works . . . 147
REFERENCES . . . 149
VITA . . . 165

PUBLICATIONS 166

LIST OF TABLES

2.1 Video properties of interest to law enforcement 43

2.2 Performance of Different Candidate Sentence Extraction Models based on Clothes Property Identification 72

2.3 Performance of Human Attribute Extraction from Text 73

3.1 Performance of YOLO for Gender and Clothes Detection in WLPD-Video-Dataset (mAP) 87

3.2 Evaluation of Human Attribute Extraction on FemmIR-text 88

4.1 Comparison of Property Identifiers for Videos with Accuracy (acc) and F1 measure on MARS dataset (%) 123

4.2 Performance of EARS and FemmIR in mAP(%) 124

5.1 Performance Comparison Results of DT2DVec 139

LIST OF FIGURES

2.1 Detailed Architecture of SKOD for “finding missing person” Use Case. 28

2.2 Simplistic view of SKOD Architecture. Left rectangle describes the streaming data ingestion module. Middle rectangle refers to a Multimodal query engine which includes data processing, feature extraction, data integration, and relevance modeling. The green rectangle denotes the data retrieval module, which includes Triggers to deliver incomplete information need over time. This would be morphed into more detailed form in later chapters. 29

2.3 Multimodal Query Engine Representation utilizing Situational Knowledge. . . . 30

2.4 Data Streaming Pipeline from Restful and Streaming tweets to applications. . . 35

2.5 Result of applying the pre-trained neural network to the Cambridge dataset. . . 37

2.6 Situational Knowledge on Demand proof-of-concept. Incoming streams of data shown in a Leaflet map. 64

2.7 An example of surveillance footage from West Lafayette, IN. 65

2.8 SurvQ video ingestion and retrieval architecture 65

2.9 The populated incident report 66

2.10 The Incident Creation Form. 67

2.11 Example actions a user might take with SurvQ. 68

2.12 The *list* and *map* viewpoints. 69

2.13 The *timeline* viewpoint. 70

2.14 Insertion Time vs Number of Trigger Functions. Insertion times are calculated on an average of 5 trials. 71

3.1 Data Fusion for Relevant Data Recommendation 82

3.2 Schema Models for Data Storage. 92

3.3 Upload Example of Data Objects. 93

3.4 Color recognition on *WLPD-Video-Dataset* 93

3.5 Tracking of a pedestrian crossing the street from a singular camera. 94

3.6 Pedestrian tracking at multiple scenes with multiple cameras. 94

3.7 Relevant Tweets with LDA in the CPAT dataset describing a *PERSON WITH GUN* in *Cambridge Area*. 95

3.8 Find-Them Demo. 95

4.1 HARG and Weak Label Generation; Left side graph refers to g^q, and the right side graph refers to g^c. Node-type labels are as follows. V: EPL Vertex, R: Root, P: Person, C: Clothes, T: Type, M: Motor-Vehicles. Squared nodes correspond to the non-empty leaf nodes. 106

4.2 Performance of FemmIR on (InciText + MARS) 125

5.1 Architecture for Weakly Supervised Joint Embedding for Multi-modal Information Retrieval. Translation module is a two-level dense video captioning model from [131]. 131

5.2 Data Information Network 132

5.3 t-SNE Embedding of Documents using DT2DVec with Random Initial Representation of Text 141

ABBREVIATIONS

SKOD	Situation Knowledge on Demand
SurvQ	Surveillance Video Querying With A Human-in-the-Loop
HART	Human Attribute Recognition from Text
EARS	Entity-Attribute-Relationship model with SQL querying
FemmIR	Feature-centric Multimodal Information Retrieval
CED	Content Edit Distance
WeSJem	Weakly Supervised Joint Embedding for Multimodal Information Retrieval
WLPD	West Lafayette Police Department
REALM	Research in Applications for Learning Machines
SAIL-ON	Science of Artificial Intelligence and Learning for Open-world Novelty

ABSTRACT

The availability of abundant multimodal data, including textual, visual, and sensor-based information, holds the potential to improve decision-making in diverse domains. Extracting data-driven decision-making information from heterogeneous and changing datasets in real-world data-centric applications requires achieving complementary functionalities of multimodal data integration, knowledge extraction and mining, situationally-aware data recommendation to different users, and uncertainty management in the open-world setting. To achieve a system that encompasses all of these functionalities, several challenges need to be effectively addressed: (1) How to represent and analyze heterogeneous source contents and application context for multimodal data recommendation? (2) How to predict and fulfill current and future needs as new information streams in without user intervention? (3) How to integrate disconnected data sources and learn relevant information to specific mission needs? (4) How to scale from processing petabytes of data to exabytes? (5) How to deal with uncertainties in open-world that stems from changes in data sources and user requirements?

This dissertation tackles these challenges by proposing novel frameworks, learning-based data integration and retrieval models, and algorithms to empower decision-makers to extract valuable insights from diverse multimodal data sources. The contributions of this dissertation can be summarized as follows: (1) We developed SKOD, a novel multimodal knowledge querying framework that overcomes *the data representation, scalability and data completeness issues*, while utilizing streaming brokers and RDBMS capabilities with entity-centric semantic features as an effective representation of content and context. Additionally, as part of the framework, a *novel text attribute recognition model* called HART was developed, which leveraged language models and syntactic properties of large unstructured texts. (2) In the SKOD framework, we incrementally proposed three different approaches for *data integration of the disconnected sources* from their semantic features to build a common knowledge base with the user information need: (i) EARS: A mediator approach using schema mapping of the semantic features and SQL joins were proposed to address scalability challenges in data integration; (ii) FemmIR: A data integration approach for more susceptible and flexible

applications, that utilizes neural network-based graph matching techniques to learn coordinated graph representations of the data. It introduces a novel graph creation approach from the features and a novel similarity metric among data sources; (iii) WeSJem: This approach allows zero-shot similarity matching and data discovery by using contrastive learning to embed data samples and query examples in a high-dimensional space using features as a novel source of supervision instead of relevance labels. (3) Finally, to manage uncertainties in multimodal data management for open-world environments, we characterized *novelties in multimodal information retrieval* based on data drift. Moreover, we proposed a novelty detection and adaptation technique as an augmentation to WeSJem.

The effectiveness of the proposed frameworks, models, and algorithms was demonstrated through real-world system prototypes that solved open problems requiring large-scale human endeavors and computational resources. Specifically, these prototypes assisted law enforcement officers in automating investigations and finding missing persons.

1. INTRODUCTION

1.1 Background and Motivation

In recent years, the proliferation of data from various sources has presented both opportunities and challenges in decision-making processes. The availability of multimodal data, which includes textual, visual, and sensor-based information, has the potential to provide valuable insights and improve decision-making in diverse domains such as law enforcement, urban planning, social applications, and public safety. However, effectively harnessing the wealth of information contained within these multimodal data sources poses significant challenges.

Traditional decision-making systems often struggle with the heterogeneity and incompleteness of data from multiple sources. The integration and extraction of decision-making information become complex due to differences in data formats, feature and annotation mismatch, and variations in data schema and types. Additionally, decision-making in open-world environments requires systems that can adapt to uncertainties, including changes in data sources, evolving user needs, dynamic or out-of-distribution data, and unexpected events. Existing literature in multimodal information retrieval or recommender systems has made progress in addressing some of these challenges by integrating data sources through correlation learning, metric learning, and autoencoders. However, these approaches have limitations in terms of domain generalization, lack of relevance labels, and restricted environments. As a result, data-driven decision making still faces significant challenges in real-world scenarios.

The motivation behind my work was to address the aforementioned challenges and enable the extraction of data-driven decision-making information in open-world environments. The ability to utilize all available and relevant information continuously obtained from multiple sources, while adapting to uncertainties, is crucial for decision-makers in various domains. By developing novel frameworks, retrieval models, and integration algorithms, we aim to empower decision-makers with accurate, timely, and context-specific information for effective decision-making processes.

To achieve this goal, we propose novel knowledge extraction framework that incorporate learning-based models and similarity metrics along with novelty characterization of multimodal information retrieval. These frameworks overcome the limitations of existing systems and provide decision-makers with a comprehensive and adaptable solution for extracting decision-making information. Our research focuses on advancing the state-of-the-art in multimodal data integration, relevance learning, scalability, and uncertainty management.

First, we propose a novel on-demand situational knowledge extraction framework that leverages streaming and RDBMS platforms and combines their functionalities in a cohesive manner. This framework addresses the challenges of heterogeneous and missing data sources, scalability, data integration, and effective data representations. In addition, we develop attribute recognition models from unstructured texts to enable fine-grained information processing and build prototypes that assist end users with their specific mission needs. Furthermore, we introduce two learning-based data integration models to overcome challenges related to approximate and ranked matching, unavailability of relevance labels, and the lack of domain experts. These models provide solutions for effective data integration in decision-making processes. Finally, we develop novel uncertainty management techniques to enhance the accuracy and adaptability of decision-making systems. By creating novelty detection and adaptation techniques for data retrieval models, our aim is to provide decision-makers with a deeper understanding of the data and enable effective decision-making in uncertain scenarios.

Through these contributions, our research endeavors to provide decision-makers with comprehensive and adaptable solutions for extracting decision-making information from diverse multimodal data sources. By addressing the challenges of heterogeneous and missing data sources, scalability, data integration, and uncertainty management, we strive to enhance decision-making processes in dynamic and uncertain scenarios.

1.2 Dissertation Contribution

This dissertation aims to accomodate the following statement:

Utilize all available, relevant information obtained from multiple sources to autonomously and continually provide decision making information that is specific to application needs, while being able to adapt to uncertainties in a open-world environment.

Through our research in this work, we have made progress towards extraction of data-driven decision making information in open-world environment from ever-increasing multi-modal data. Our main contributions fall into the following categories.

1.2.1 Decision Making Information Extraction from Multimodal Data

In this work, our focus was to tackle the challenges of heterogeneous and missing data sources, while discovering appropriate representations of the content and context for the data integration task. To that end, we proposed a novel framework for situational knowledge extraction and dissemination with high level semantic features for the content and context representation and as an input to a multimodal query engine (Section 1.2.1.a). We proposed three different data integration approaches as part of the multimodal query engine for knowledge modeling (Section 1.2.2). We augmented our proposed framework with a novel attribution extraction model for unstructured text as part of the multimodal query engine (Section 1.2.1.c), while building a real-world application prototype for police detectives to help conduct investigations (Section 1.2.1.b).

1.2.1.a Situational Knowledge Query Engine Framework

First, we proposed a novel framework for the delivery of multimodal decision making information from all available and incoming data sources, solving the challenges of scalability and data completeness of real-world applications. Our contributions are as follows.

1) SKOD is a *scalable, real-time, on-demand* situational knowledge extraction and dissemination framework that processes streams of multi-modal data utilizing publish/-subscribe stream engines and a multimodal query engine for data integration and relevance learning using semantic features.

2) SKOD proposes using *high level semantic features* to represent heterogeneous source content and mission context for multimodal data recommendation.

3) Novel multimodal query engine that continuously builds a multi-modal relational knowledge base using SQL queries.

4) SKOD pushes dynamic content to relevant users through triggers based on modeling of users' interests, solving the issue of *incomplete data and information need over time*.

5) We implemented a prototype of the proposed framework solving the use-case of *Urban Information System* with dataset collected from Cambridge, MA.

1.2.1.b Novel Use Case for West Lafayette Police Department

We worked with the West Lafayette police department to solve a novel use-case of *Surveillance Video Querying Engine with Human-in-the-Loop*, using our proposed Situational Knowledge Query Engine. Our contributions are as follows.

1) Novel system prototype for *Surveillance Video Querying Engine with Human-in-the-Loop* with Police Department from Chesterfield, NH and West Lafayette, IN.

2) Evaluation of scalability capabilities of the proposed knowledge extraction framework.

3) We proposed augmentation techniques on top of existing object detection frameworks for video feeds and images for fine-grained system properties, while tackling the challenges of poor video quality and low data availability.

1.2.1.c Novel Human Attribute Recognition model

To solidify the SKOD framework, we proposed a novel human attribute recognition model from large unstructured text which leverages pattern-matching techniques and contextualized language models while exploiting the syntactic grammatical properties to extract properties describing a person. This model can be generalized for any object properties, ensuring a granular level text property identifier for any mission. Our contributions are described as follows.

1) To the best of our knowledge, this is the first work to explicitly investigate human attribute extraction from the large unstructured text.

2) We formally define the task of *Human Attribute Recognition from Text* based on the interviews with Police Department. We further divide it into two subtasks, and propose a novel algorithm for discovery of both the attribute name and the value.

3) In the process, we introduced a fully annotated novel dataset derived from real life and synthetic investigation reports and press releases.

4) Experiments show our proposed approach performs better than standalone language models for fine-grained attribute detection.

1.2.2 Data Integration and Relevance Learning

After the data preparation and feature learning, the next challenge in data-driven decision making arrives from data integration. Data integration from various sources suffers from heterogeneity issues from differences in feature names that hold similar data, annotation mismatch, and variations in data schema and types. Existing multimodal information retrieval approaches suffer from scalability and domain-specificity issues. To solve that, we proposed EARS, using a mediator-based approach and semantic mapping to connect disconnected data sources through their features in Section 1.2.2.a. Although this can scale upto petabytes of data due to RDBMS capabilities, it faces issues when some application domains require approximate matching and ranked results from all incoming data. To tackle that, we proposed FemmIR where we learned a coordinated graph representation from the semantic features of the input and query data samples, which maintains the dissimilarity between data objects as a learning objective (Section 1.2.2.b). Finally, to solve the issue of lack of labeled data, we propose WeSJem, a weakly supervised joint embedding model that maps the data samples and their associated features in a manner that maintains the similarity relationship based on the features (Section 1.2.2.c).

1.2.2.a Applying Schema Mapping and Data Fusion to "Missing Person" Problem

In this paper, we make the following contributions:

1) Proposed system delivers integrated query results over time using a mediator approach along with Postgres triggers.

2) A semantic mapping is employed between the mediated schema and the data sources to query the limited properties-of-interest in real-world applications.

3) The original query-by-example is translated into conjunctive queries among data sources and a SQL-Join on the task-specific features is performed at run-time to integrate all the relevant sources to the query example.

1.2.2.b Feature-centric Multimodal Information Retrieval with Graph Matching

In this paper, we make the following contributions:

1) Novel multi-modal information retrieval approach to find multi-media data relevant to information need expressed as Query-by-Example and Query-by-Properties.

2) The approach leverages a neural-network based graph-matching technique to capture the interactions between the query example and the streaming data features, with a weak supervision from a novel distance metric for data samples.

3) Novel edit distance metric, *CED*, to measure the amount of difference between two data samples based on their semantic features.

4) We evaluate FemmIR on a real-world application for Missing Persons, with an unannotated dataset and a property-specific information need. We demonstrate that FemmIR shows similar performance to other retrieval systems [1] while leveraging pre-identified properties, on a novel multi-media dataset comprised of pedestrian identification and real-world dataset.

5) Additionally, we benchmarked property identifiers for the visual modalities to identify the best model for the downstream retrieval task.

1.2.2.c Weakly Supervised Joint Embedding for Multimodal Information Retrieval

Our contributions are summarized as follows:

1) Novel multi-modal joint embedding model which is pre-trained with weak supervision from semantic features. The model can use existing video and image translation models along with existing text feature extractors. WesJem can be applied to any application domain for cross modal retrieval.

2) The multi-task joint objective function is built upon a data information network based on how different data samples interact with each other via their structural features.

3) WesJem has the flexibility to take into account any user provided features and similarity labels during the joint multi-task training. This allows it to be adapted by application domains which already have extracted features.

4) Preliminary experiments using only *topics* as features demonstrate our models effectiveness on retrieval and similarity evaluation tasks.

1.2.3 Uncertainty Management in Multimodal Information Retrieval

In static environments, AI systems can follow rigid rules or past experiences in order to execute certain tasks. However, in dynamic, uncertain real-world environment, unexpected changes (i.e., novelties) can occur, and the AI system is expected to detect and adapt to these changes in a timely manner. We need to characterize novelties in AI systems to detect novelties and act accordingly in non-stationary environments.

To that end, we characterize and formalize novelties in multimodal information retrieval task in terms of data shift. Proposed framework includes a novelty detection and response module in terms of our weakly supervised retrieval model, WeSJem. For that, we proposed a pre-training strategy for handling out-of-distribution inputs with a modular approach. To the best of our knowledge, this is the first framework that formalizes novelty for multimodal retrieval task.

1.3 Dissertation Organization

The rest of this dissertation is organized as follows: Chapter 2 presents a novel framework for open-world data-driven decision-making information retrieval, while discussing how the framework handles the challenges of heterogeneous data representation, on-time data delivery, storage, and data incompleteness. Section 2.3 introduces a novel attribute detection model for unstructured text, and section 2.2 introduces augmentations on perception domain object detection models for open-world decision-making purposes. Section 2.2 also showcases how we implemented a real-world societal application using our proposed framework.

Chapters 3 – 5 discuss three novel data integration approaches that utilize semantic features as content representation. Chapter 3 proposes EARS, which relies on semantic mapping and SQL-JOIN to achieve a scalable and exact data integration model. Chapter 4 solves the issue of approximate matching by employing a neural network-based graph-matching approach to perform relevance matching between data samples. Chapter 5 finally proposes a weakly supervised representation learning model to solve the lack of annotation problem, by automatically mapping the data samples in a higher embedding space while maintaining the similarity constraints between data samples in terms of their features.

Chapter 6 discusses how we can handle uncertainties in an open-world environment for multimodal information retrieval. Finally, chapter 7 concludes the dissertation with a summary of the main contributions and a discussion of our future plans to complete the AI life cycle for multimodal data management to perform data-driven decision-making.

2. DATA-DRIVEN DECISION MAKING INFORMATION EXTRACTION FROM MULTIMODAL DATA

2.1 Situation Knowledge on Demand (SKOD)

The past decade has witnessed an unprecedented *volume* of data being generated by a *variety* of sources at very high *velocity*, resulting in the rise of the *big data* paradigm. Specifically, the developments in social networks and Internet of Things (IoT) have created a plethora of multi-modal data sources that generate billions of data records every second, only a small fraction of which readily translates into useful information. While the availability of such vast amounts of data has made it possible to build large knowledge bases, on-demand extraction of highly relevant situational knowledge for specific missions from those heterogeneous data clouds remains a difficult task for the following reasons:

1. Accurate correlation of data from different resources for billions of data items is a daunting task;

2. A knowledge base built upon a specific ontology may not cater to the needs of a mission when additional mission requirements/user interests are defined later;

3. The storage of the most relevant data in the knowledge base is essential to avoid performance degradation with growing data;

4. Generalization of knowledge bases irrespective of mission needs is a challenge.

Many critical missions will require real-time targeted dissemination of information to interested parties as information becomes available. Achieving high-performance, accurate information extraction and propagation requires

1. accurate modeling of the different users' interests;

2. application of intelligent filters on streaming data to capture and correlate the most relevant aspects;

3. triggers for communicating the gathered information to the interested parties.

In this section, we propose SKOD, a framework for situational knowledge on demand, which provides high-performance real-time analysis of streaming data of multiple modalities from different sources to dynamically and continuously build mission-specific knowledge bases for assistance in decision making. In order to capture data most relevant to user needs, SKOD uses past user query patterns to construct the knowledge base.

Our approach provides a scalable solution for modeling different user interests over vast amounts of data while allowing flexibility for future incoming data. Additional interests can immediately be integrated by defining new queries on the knowledge base. SKOD currently handles pattern extraction from streaming video and text data, but the extensible architecture allows facile integration of additional data modalities such as audio, sensor data, signals, and others.

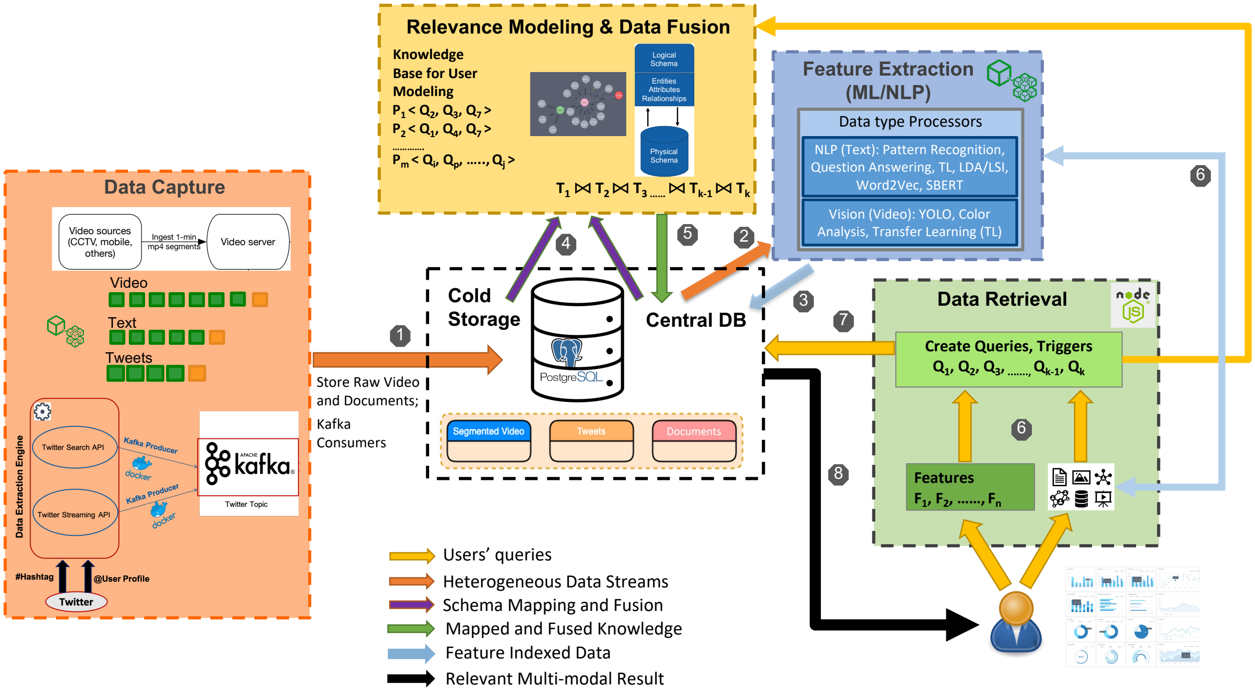
2.1.1 Example Application Scenario

In order to clearly illustrate the objectives and operation of SKOD, we describe an example application scenario of the system in this section. Let us consider a city information system, which provides access to multiple agents (e.g., police, public works department, citizens, emergency personnel, homeland security) with varying missions, hence varying information needs. In such a system, while the police would be interested in patterns such as unsafe lane changes, locations visited by a suspicious person, to name a few; the public works department would be interested in patterns such as potholes and occluded street signs. An example query to be submitted to this system by a police officer is:
Q1: *List cars parked next to fire hydrants illegally today.*

To answer Q1, we will require detecting cars and fire hydrants in video frames and tweets, given the available data sources are city surveillance cameras and Twitter. The query response will provide information that the policeman will always be interested in, therefore as new data streams in, patterns matching the query should be communicated to the policeman and other police officers as well, due to the similarity of their profiles to the user submitting the query. A different user of the system (a firefighter) can later submit **Q2:** *Get locations of leaking fire hydrants.* While this query will be able to utilize the knowledge base created in response to Q1, it will build upon it to find patterns of the act *leak* in both data sources as they stream additional data to the system.

2.1.2 SKOD System Architecture

The SKOD architecture consists of three large modules - 1) streaming platform to handle the vast amount of heterogeneous incoming data, 2) multi-modal query engine to model the user interest based on their previous queries and 3) the front end with the indexing layer. In Figure 2.2, we show an overview of SKOD's architecture. We will describe a more detailed overview of the architecture for "finding missing person" use case in Section 3.3, as shown in Figure 2.1. We describe the three primary modules below.



Relevance Modeling & Data Fusion
Knowledge Base for User Modeling
$P_1 < Q_2, Q_3, Q_7 >$
$P_2 < Q_1, Q_4, Q_7 >$
....
$P_m < Q_i, Q_p, ..., Q_j >$
Logical Schema
Entities Attributes Relationships
Physical Schema
$T_1 \bowtie T_2 \bowtie T_3 ...... \bowtie T_{k-1} \bowtie T_k$
Feature Extraction (ML/NLP)
Data type Processors
NLP (Text): Pattern Recognition, Question Answering, TL, LDA/LSI, Word2Vec, SBERT
Vision (Video): YOLO, Color Analysis, Transfer Learning (TL)
Data Capture
Video sources (CCTV, mobile, others)
Ingest 1-min mp4 segments
Video server
Video
Text
Tweets
Data Extraction Engine
Twitter Search API
Twitter Streaming API
Kafka Producer
docker
Kafka Producer
docker
APACHE kafka
Twitter Topic
#Hashtag
@User Profile
Twitter
1
Store Raw Video and Documents;
Kafka Consumers
2
3
4
5
Cold Storage
PostgreSQL
Central DB
Segmented Video
Tweets
Documents
Data Retrieval
node JS
7
Create Queries, Triggers
$Q_1, Q_2, Q_3, ..., Q_{k-1}, Q_k$
6
Features
$F_1, F_2, ..., F_n$
6
8
Users' queries
Heterogeneous Data Streams
Schema Mapping and Fusion
Mapped and Fused Knowledge
Feature Indexed Data
Relevant Multi-modal Result

Figure 2.1. Detailed Architecture of SKOD for "finding missing person" Use Case.

2.1.2.a Streaming Broker

Due to the latency-sensitive set of applications that SKOD aims to tackle to consume data from heterogeneous sources, this work relies on Apache Kafka to expose a real-time

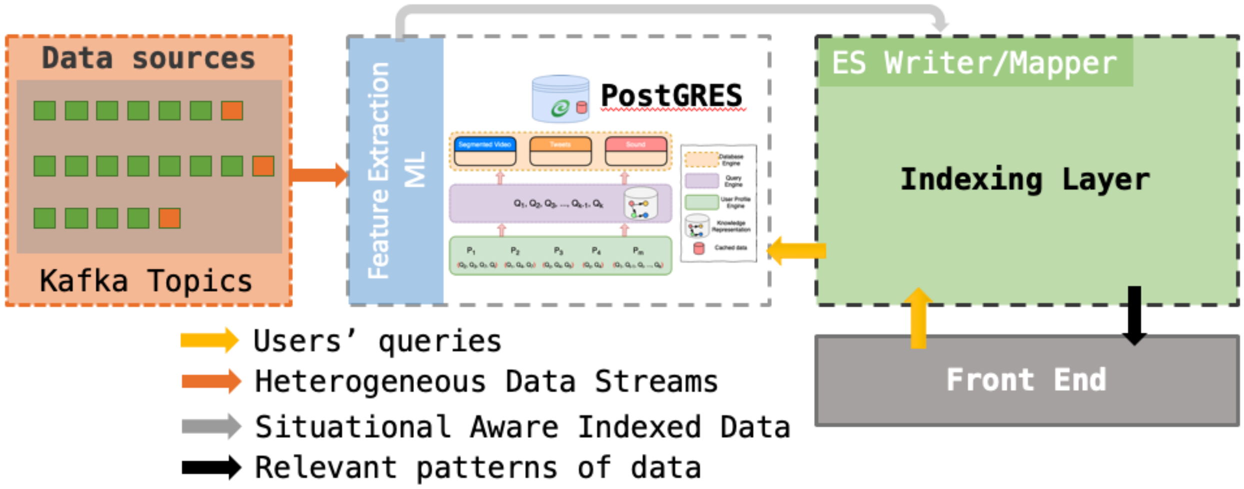


Figure 2.2. Simplistic view of SKOD Architecture. Left rectangle describes the streaming data ingestion module. Middle rectangle refers to a Multimodal query engine which includes data processing, feature extraction, data integration, and relevance modeling. The green rectangle denotes the data retrieval module, which includes Triggers to deliver incomplete information need over time. This would be morphed into more detailed form in later chapters.

stream processing pipeline. Apache Kafka is a scalable and fault-tolerant publish-subscribe messaging system. Kafka achieves the capability to store and process data from multiple applications (producers) through a topic abstraction system. As an output, multiple applications can consume the inserted data from all the producers asynchronously and without any loss. The producers/consumers abstraction allows SKOD architecture to provide real-time analysis and recommendation capability. Apache Kafka features allow to store the raw incoming data in Postgres and consume the same data by text and video processing applications simultaneously.

Currently SKOD architecture consumes both RESTful, and streaming data from Twitter, incident Reports, image and video feeds through Kafka. SKOD is capable of integrating data from other real-time applications (i.e., sensor, audio, files, JDBC) through Kafka Clients or Kafka Connect. Kafka Clients allow to pass and retrieve messages directly to and from Kafka as long as the message can be converted to bytes. We show a detailed view of SKOD data streaming pipeline in Figure 2.4 for different types of Twitter data.

2.1.2.b Multimodal Query Engine

The multimodal query engine consists of several sub-modules. The query engine also accommodates the unit for feature-analysis of heterogeneous data for identifying personalized events. The first sub-module consumes the streams of data provided by the *streaming broker* and stores them directly in the relational database (Postgres). The second sub-module extracts features from each mode of data with a separate processing unit. For our current implementation, we focus on processing video and unstructured text to extract common features to most domains. In the final module of the multi-modal query engine, SKOD utilizes users' SQL queries to build the knowledge base on top of a relational database and pushes relevant content to users without user intervention. In Figure 2.3 we observe the structure of the query engine.

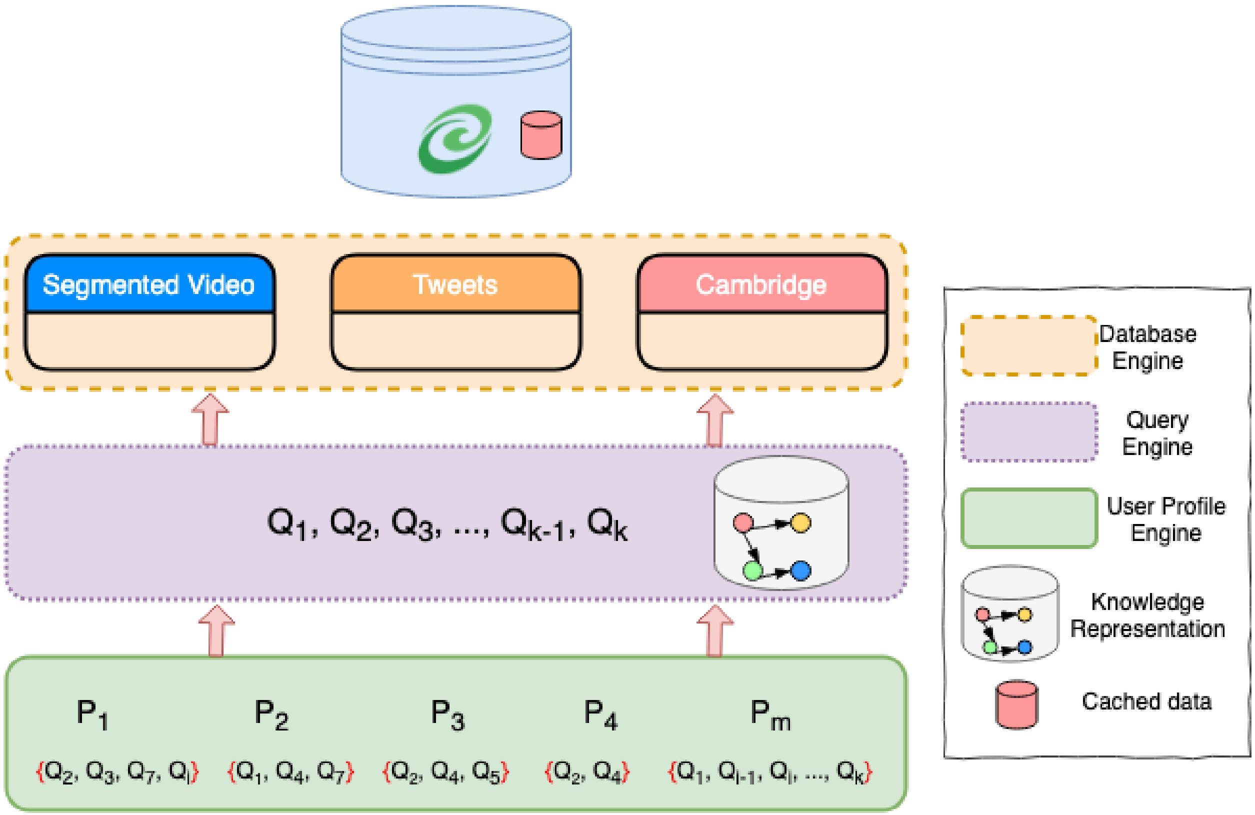


Figure 2.3. Multimodal Query Engine Representation utilizing Situational Knowledge.

Feature Extraction from Image and Video Streams

Video data represents a separate and unique modality in the SKOD multi-modal system for storing and extracting knowledge on demand. Video data comes in large amounts, unstructured, and raw video is unlabeled, frequently in need of processing, cleaning, and preparing for the next stage in the data flow.

Video can be viewed as a sequence of frames, where each frame is characterized by its bitmap that can later be transformed into a multidimensional array or a tensor. The need to work with extensive digital representations requires specific ways of storing and operating with the video data, which are different from those of text and structured data. When the knowledge must be extracted efficiently on demand from the heterogeneous multi-modal database, there are several challenges to be resolved: (1) Entities from each frame have to be accessible for user queries, user-defined stored procedures, and event triggers; (2) For connecting with other modalities in a poly-store environment, these entities must be stored in a way that they can be matched with the text data and text metadata as well as data from other modalities for further analysis; (3) There must be a way to obtain entities in an ad-hoc manner to extract knowledge from streams of video. We resolve these challenges utilizing two off-the-shelf solutions: Apache Kafka for streaming video in a scalable, fault-tolerant manner and the YOLO real-time object detection system [2].

In Section 2.2, we proposed augmentation techniques for video and image feature extraction methods for more granular level features or properties for specific mission needs.

Feature Extraction from Text

Understanding unstructured texts has always been a daunting task. Even with the recent rise of language models it is hard to parse unstructured social texts into domain-independent features.

We first preprocess the text using Stanford CoreNLP [3], extract named entities and co-reference chains. Then we create a separate table in Postgres to save each tweet with its associated named entities i.e, LOCATION, ORGANIZATION, PERSON, saving them as text arrays and associated topic with the tweet. Further, we create another column `objects`, which are any words in the tweet except stop words and the ones identified in named entities.

In Section 2.3, we proposed a novel feature extraction method from unstructured text for more granular level features for more specific mission needs.

Data Integration and Knowledge Modeling

Unified knowledge representation for all streamed data is required for the query engine to extract useful knowledge and disseminate information to relevant parties efficiently. In SKOD, we represent knowledge using relational data and SQL queries on the data, which persist for the lifetime of the knowledge base and grow with additional user interests. Representation of textual data such as tweets and online news is more straightforward through the extraction of topics and keywords; which can directly be entered into the corresponding columns in the RDBMS tables. Multimedia data such as video and audio are represented both with the binary data and the text data extracted as a result of the processing performed on the binary data. The stored data also includes available metadata for all modalities, such as timestamp, geolocation, and some others. The metadata is especially useful when correlating multiple forms of data for the same events.

The schemas of the PostgreSQL tables storing the extracted features from the tweet text and video frames are as follows:

```
TWEETS(tweet_id INT,
    locations VARCHAR(100)[],
    objects VARCHAR(100)[],
    organizations VARCHAR(100)[],
    persons VARCHAR(100)[],
    dates VARCHAR(100)[],
    times VARCHAR(100)[],
    topic VARCHAR(100)[]
    created_at DATE)
```

```
VIDEO_FRAMES(video_id INT,
    frame_id INT,
    locations VARCHAR(100),
    objects VARCHAR(100)[],
    people VARCHAR(100)[],
    timestamp DATE,
    image BYTEA)
```

Here locations, organizations, and persons are different classes of named entities and other classes can be defined as necessary. Typical attributes are used to facilitate joins between the tables for data correlation. Attributes in different tables may have different names, but have commonalities, i.e., timestamp and created_at, or people and persons. Given the initial knowledge base is built upon **Q1** mentioned in Section 2.1.1, new streams of video data will result in running the object detector for cars and fire hydrants, and the extracted data will be inserted into the database. Similarly for streaming Twitter data, tweets that have the objects car and fire hydrant will be inserted into the relevant table.

Q1 for a system with these two data sources will translate into multiple SQL queries for the situational knowledge query engine:

```
SELECT video_id, frame_id
FROM VIDEO_FRAMES
WHERE ‘car’ = ANY(objects)
AND ‘fire hydrant’= ANY(objects)
```

```
SELECT tweet_id
FROM TWEETS
WHERE ‘car’ = ANY(objects)
AND ‘fire hydrant’= ANY(objects)
```

```
SELECT t.tweet_id, v.video_id, v.frame_id
FROM TWEETS t, VIDEO_FRAMES v
WHERE ‘car’ = ANY(t.objects) AND ‘fire hydrant’ = ANY(t.objects)
AND ‘car’ = ANY(v.objects) AND ‘fire hydrant’ = ANY(v.objects)
AND v.location = ANY(t.locations)
```

As data from either resource is streaming in, patterns matching these queries will create triggers for relevant data to be communicated to interested users. Note that the complete system requires translation of natural language questions into SQL queries through entity recognition, and constructs for creating all related queries given the tables for different data sources and their common attributes. Although this initial design is limited to recognition of objects, a richer knowledge base will require incorporation of activity recognition in videos and tweets. In chapters 3 – 5, we described different data integration approaches for more complex multimodal data matching and knowledge modeling.

2.1.2.c Data Retrieval and Missing Information Completion

SKOD includes fixed queries on data streams from multiple sources, both separate and combined. The queries are then stored to build the knowledge base, which in return models the user interests. SKOD can provide users with information similar to their previous queries as well as missing information on their existing information. This information is delivered to the user using trigger events in the relational database. Similar queries and repeated accesses to similar data are cached to provide better throughput. The front-end queries an indexing layer based on Lucene indexes to improve throughput.

Elasticsearch is a distributed indexing and search engine. SKOD queries Elasticsearch through a RESTful API. Moreover, Elasticsearch utilizes Lucene indexes under the hood. Naturally, Elasticsearch achieves fast response times because it queries an index instead of querying text or video directly. The basic structure is called *Document*. Elasticsearch returns a ranked set of results according to the relevance of the query. SKOD uses Elasticsearch to rank relevant content to push to the end user.

2.1.3 Prototype Implementation

2.1.3.a Dataset

For unstructured text, SKOD used tweets in this prototype. There are two types of tweet data available for scraping - RESTful data (historic data) and streaming data. SKOD uses Twitter search API to collect RESTful data and Twitter streaming API for collecting real-time tweet streams. It creates independent docker containers for the producers, which can take tags and timelines as environment variables and run simultaneously. Since there can be overlap of tweet data from multiple producers, SKOD uses the Kafka streaming platform to handle the asynchronous, scalable and fault tolerant flow of tweets using the same topic abstraction for all. After the data is in Kafka, SKOD uses two separate consumers - 1) to parse and populate Postgres with the tweet and associated metadata, and 2) to pass the raw tweets to a feature extraction engine. Figure 2.4 shows an overview of the architecture.

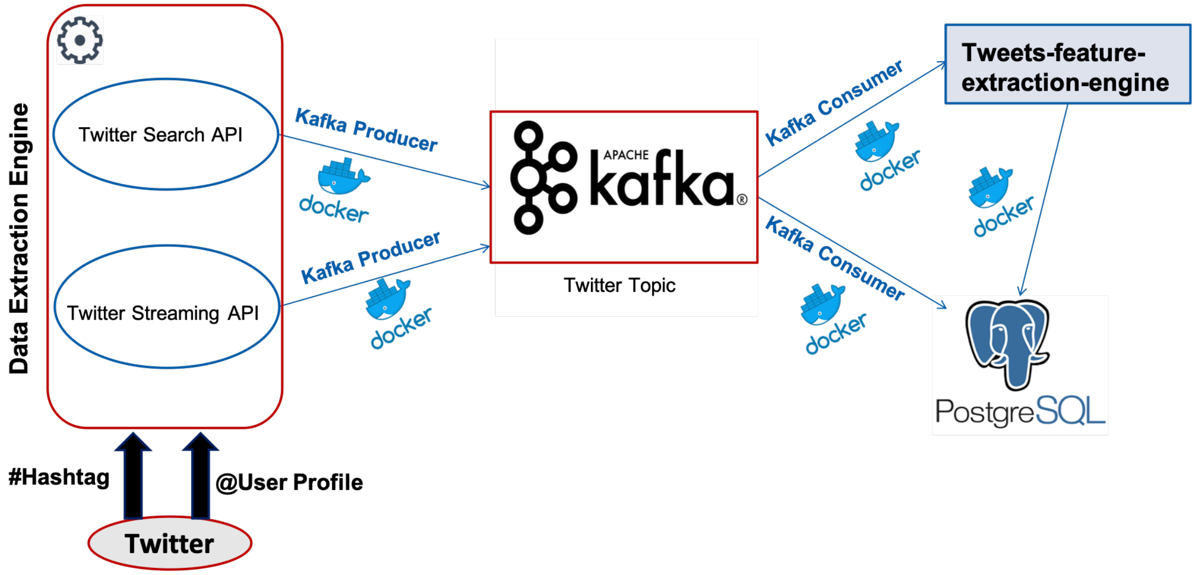


Figure 2.4. Data Streaming Pipeline from Restful and Streaming tweets to applications.

Since the city of Cambridge was the point-of-focus for the data used in this work, the target was to collect a million tweets that discuss events and entities in Cambridge, MA along with all the metadata from Twitter. Twitter data can be collected by hashtags, user timelines, geo-data, and general queries. In SKOD, we chose to search by hashtags and user timelines. For that purpose, about 15 hashtags and 15 user timelines were manually selected after going through profiles in timelines and descriptions for hashtags. For example, @CambridgePolice warns about any possible crimes or street law changes, while @bostonfire talks about fire-related incidents in Boston. At a much broader scale, hashtags like #CambMA include all tweets by many Cambridge, MA departments. For video dataset, we used dashcam videos from Cambridge, MA which were collected by uber drivers.

2.1.3.b Experimental Settings

For the implementation of twitter APIs, SKOD uses tweepy.api[1]. There is a class method API() which allows to search by both hashtags and timelines by providing a wrapper for twitter APIs. The Twitter streaming API is used to download twitter messages in real time. In Tweepy, an instance of tweepy.Stream establishes a streaming session and routes messages to StreamListener instance by allowing a connection to twitter streaming API.

High-level Feature Extraction from Text and Video

At the time of the implementation, we had around 80K tweets in Postgres. More were being accumulated as the module keeps running. The consumers inherit twitter data as JSON messages. The JSON message is parsed to extract relevant metadata. Different types of tweets are identified, i.e., original, retweet, and quoted tweets. The tweet text with all the parsed metadata along with the original JSON message is saved in Postgres. With the tweet text, we obtain a social network connected by retweets and follows.

We ran the pretrained 7 class NER CRFs from Stanford toolkit [3] to identify the entities. For topic extraction, SKOD uses the Latent Dirichlet Allocation (LDA) method [4]. We show the schema of the PostgreSQL table storing the extracted features from the Tweet text

[1] ↑tweepy

in section 2.1.2.b. SKOD wraps the producers and consumers in docker containers. The producers and consumers take the Kafka hostname and port number as input, along with the tags and timelines in files.

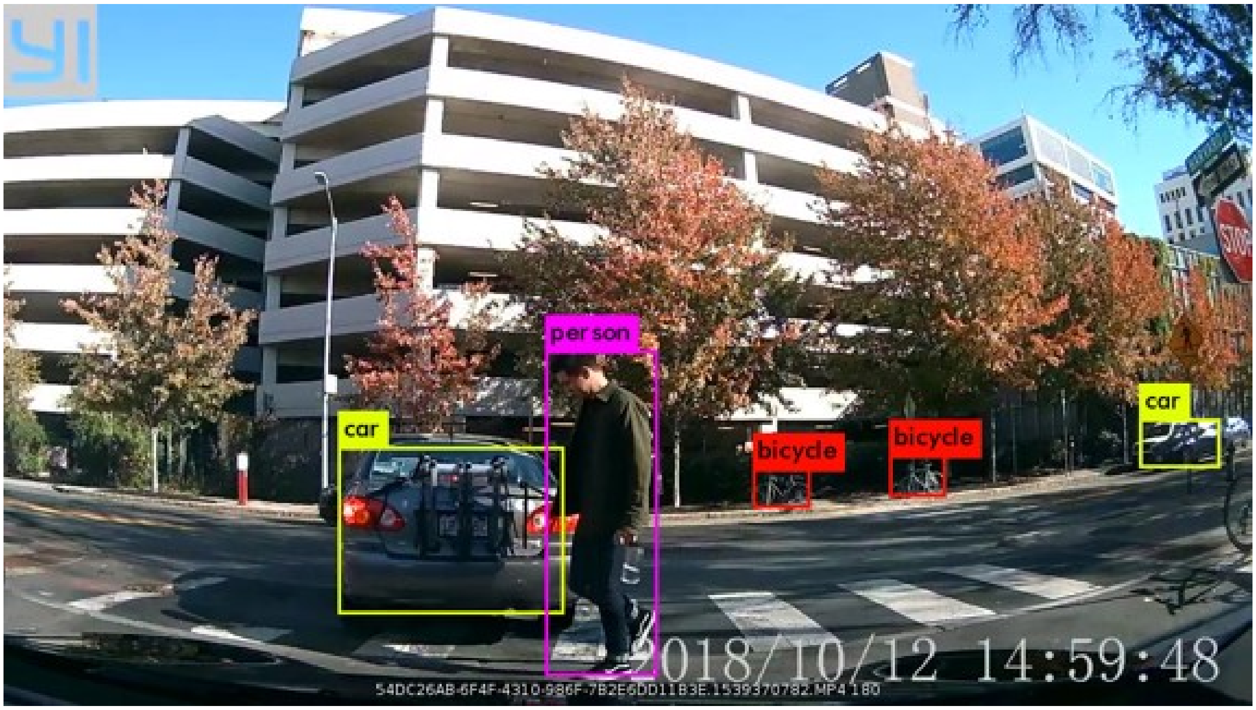


Figure 2.5. Result of applying the pre-trained neural network to the Cambridge dataset.

In the prototype implementation, SKOD uses YOLO [2], a universal pre-trained neural network as a tool for object extraction and recognition in the video data. SKOD's video processing feature differentiates between 150 object classes. SKOD identifies the objects in the video on a frame-by-frame basis. The result obtained for a particular video frame in the collected Cambridge dataset using YOLO is shown in Figure 2.5. For each processed frame, the recognized data and metadata are stored in the RDBMS and can be used for queries that involve the video data modality.

2.1.3.c Front End and Indexing Layer

The front-end utilizes React[2], which is a JavaScript library for building user interfaces. Also, we manage states and side effects using the Cerebral[3] library. We leverage interactive maps via the Leaflet[4] library integrated with React and Cerebral. SKOD caches the most frequent queries to provide faster response times. SKOD's architecture comprises a set of Node.js and python microservices, i.e., Docker containers. In Figure 2.6, we demonstrate the integration of multimodality combining the extracted Twitter data with the front-end (we utilize GPS coordinates in the Twitter data in GeoJSON format to render the Twitter data in the Leaflet map). The Tweets come through the Apache Kafka broker. Then the data is stored in the backend (Postgres). Finally, the Web application queries the indexing layer and it also watches for new changes utilizing WebSockets SKOD provides an additional layer of cache storing content in the browser using PouchDB[5] similar to the OADA cache library[6]. SKOD future releases include the creation of an elastic cache-layer building a rich set of network topologies on the edge of the network utilizing Web Browsers with Real-Time Communication (WebRTC[7]) [5].

2.1.4 Related Work

The rise of the big data paradigm in the past decade has resulted in a variety of approaches for processing and fusion of data of multiple modalities to extract useful knowledge. Poria et al. proposed an approach for fusing audio, visual and textual data for sentiment analysis [6]. Foresti et al. introduced a socio-mobile and sensor data fusion approach for emergency response to disasters [7]. Meditskos et al. developed a system for multi-modal fusion of data including language analysis results, and gestures captured from multimedia data streams to provide situational awareness in healthcare [8]. Adjali et al. proposed an approach for multi-modal fusion of data from sensors to provide ambient intelligence for robots [9]. While

[2] ↑https://reactjs.org/
[3] ↑https://github.com/cerebral/cerebral
[4] ↑https://leafletjs.com/
[5] ↑https://pouchdb.com/
[6] ↑https://github.com/OADA/oada-cache
[7] ↑https://webrtc.org/

successful for the specific domains considered, these approaches may not generalize to other domains.

One application of multi-modal data fusion that has gained increasing interest is visual question answering. Zhu et al. [10] tackle the visual question answering problem by building an external knowledge base via iterative querying of the external sources. Their system uses a neural approach where task-driven memories are actively obtained by iterative queries and produces the final answer based on these evidences. Although they take a query based approach for the QA task, their data source is just limited to images. Our approach aims to build a knowledge base integrating visual, textual, and structured data along with the relations among them.

Likewise, Wu et al. propose a method combining an internal representation of image content with information from an external knowledge base to answer complex image queries [11]. Video analytics represents a class of problems related to one of the dimensions of multi-modal systems exploration, namely efficient and fast video querying. In [12], the authors develop a declarative language for fast video analytics and enhance it with the engine that accepts, automatically optimizes and executes the queries in this language efficiently.

While many multi-modal knowledge bases are constructed using learning-based data fusion approaches on large static datasets, query-driven approaches construct knowledge bases through repeated querying of text and multimedia databases. Nguyen et al. [13] propose QKBfly, an approach for on-the-fly construction of knowledge bases from text data driven by queries. QKBfly utilizes a semantic-graph representation of sentences through which named-entity disambiguation, co-reference resolution and relation extraction are performed. Bienvenu et al. propose an approach for utilizing user queries and the associated user feedback to repair inconsistent DL-Lite knowledge bases [14]. The constructed knowledge bases will in most cases include inconsistencies and missing information. Probabilistic knowledge bases have been introduced to handle these inconsistencies by assigning belief scores to facts in the knowledge bases [15], [16], followed by approaches to fuse data from multiple probabilistic bases [17].

Traditional knowledge bases are used for information extraction to answer user queries as they are submitted. On the other hand, dynamic detection of events on streaming data is important for many systems today, due to the need to make users aware of important events in real time. This has resulted in the development of complex event processing systems for purposes such as crisis management [18], to create triggers when streaming data matches pre-defined patterns [19]. Although these systems provide real-time event notification to interested parties, their rule base in most cases is fixed, not supporting evolving mission requirements and users with different interests.

2.1.5 Conclusion and Future Work

In this paper we proposed SKOD, a situational knowledge on demand engine that aims to provide a generic framework for dynamically building knowledge bases from multi-modal data to enable effective information extraction and targeted information dissemination for missions that might have evolving requirements. In order to provide the best run-time performance and accuracy, SKOD uses a query-driven approach to knowledge base construction. Being query-driven, it is expected to enable effective information retrieval and dissemination in a variety of fields including law enforcement, homeland defense, healthcare etc., all building knowledge upon the specific interests of the system users.

The development of SKOD is in progress with components for stream data processing, feature extraction from video and text data currently in place. Our future work will involve the development of components for query processing, user similarity modeling, and user relevance feedback to achieve highly accurate real-time targeted information propagation. The system will be evaluated with multiple rich multi-modal datasets such as Visual Genome [20], COCO [21], YouTube-8M [22], and collected tweets and video data set of our own for various missions and user types.

2.2 Surveillance Video Querying With A Human-in-the-Loop (SurvQ)

2.2.1 Introduction

Urban areas have many video cameras continuously recording some field of view. There are fixed cameras on light poles, cameras on city vehicles, and cameras on police personnel. In addition, there are surveillance cameras on private property, NEST doorbells, interior spaces, and many private vehicles. The resulting video streams are crucial for police officers when solving a range of everyday crimes, but often entail hours of tedious manual examination, thereby wasting scarce police resources that could be put to better use.

MIT and Purdue researchers have developed a Surveillance Video Querying system, SurvQ, for surveillance video information. The Chesterfield, NH police department is assisting in providing mission requirements. The West Lafayette, IN police department has provided video data from cameras in the downtown area plus redacted police dispatch reports. The SurvQ prototype must operate at sufficient scale to handle the West Lafayette use case (about 200 fixed and mobile cameras), and with an analysis loop that gives rapid answers to busy law enforcement officers. SurvQ has been designed so it is readily adaptable to larger deployments and other surveillance applications, such as those found in securing military bases and disaster recovery.

The Detective In the Loop – The primary focus of SurvQ is to assist humans performing analytical tasks: that is, police detectives solving crimes and tracking individual suspects. A typical scenario is an incident report of the form assault reported at time XXX in location YYY. Suspect is of medium height, wearing jeans and a baseball cap. In this particular use case from West Lafayette, a suspect matching the description was spotted on a public bus camera, and that led to his arrest. At the present time, examining video footage is a manual process for police, and takes hours and hours of time. The request from both police departments is simple: please help us be more efficient at searching video to track suspects. The general use case is to find video frames matching a given description in a given geographic area and time range. Detectives want both an off-line system to search historical data and a real-time (standing query) facility with short response time (under 60 seconds). An example video frame from a West Lafayette street is shown in Figure 2.7. In addition to accepting video queries via direct human input, one way to make detectives more effective would be to draw query specifications from multiple fused sources, such as tweets or hand-written police reports.

2.2.1.a Surveillance Video and Detective Use Cases

Surveillance video has a collection of notable characteristics. First, it is often fairly low resolution and therefore difficult to process with sophisticated techniques. Second, the lighting is usually poor, because of glare, night conditions, fog, rain, or snow. Third, interesting objects are often in the background, often facing away from the camera, and are usually closer to thumbnails than images. Fourth, interesting objects often have properties that are rarely observed, so obtaining training data may be a challenge for building machine learning (ML) systems.

The West Lafayette Police department shared the 31 object properties they are most interested in, and these are shown in Table 2.1. Several things should be noted about this property set.

Some properties are relatively rare. For example, smoking is rare on college campuses and sandals are very rare in December in Indiana. Trying to find rare events by using traditional machine learning techniques that depend on obtaining large amounts of training data is not likely to succeed.

Some properties are visually very small. Tattoos in surveillance video are a few pixels. Of course, this makes recognition difficult.

The set of interesting properties for the police application is surprisingly modest. To meet their needs, it is easy for computer scientists to hypothesize all the possible data values that might be extracted from an image. Also, police queries — as we saw in interviews with police officers and by examining police reports — are driven by a relatively small number of query types. We believe this fact can help both in designing an efficient system and in addressing citizens privacy concerns.

Design Considerations – Training data is challenging. In an early experiment we demonstrated that building a classifier to find people wearing jeans by training on high quality web images failed to recognize jeans in surveillance video. Hence, transfer learning may not work well. In addition, many of the properties in Table 2.1 are subjective, and humans may differ on whether they are present. In other words, the input is very noisy.

We face problems of scale. West Lafayette has more than 100 cameras. The administration wishes to retain all video for months. This quickly becomes a terascale to petascale problem.

Traditional deep learning is expensive at scale. We anticipate there will be additional properties of interest to the police beyond those in Table 2.1. As we have demonstrated, our transfer learning attempt failed to find jeans. Building a training set for the properties of Table 2.1 is a tedious manual process. In addition, model runtime at scale is costly. It is not clear that a deep learning solution is affordable by the City of West Lafayette.

Table 2.1. Video properties of interest to law enforcement

White	Black	Hispanic	Asian
Male	Female	Tattoos	Beard
Bald	Hair color	Sandals	Shoes
Boots	Jeans	Pants	Shorts
T-shirt	Baseball hat	Jacket	Tall
Shorts	Walking	Running	Motorcycle
Bicycle	Truck	Passenger car	Skateboard
Smoking	Backpack	Headphones	

Overall Approach – SurvQ applies property recognizers to the video streams and loads both video and the properties into a Postgres database [8]. To achieve scalability, it can be optionally be loaded into Citus [9], a parallel multi-node terascale extension of Postgres. The detective spends his or her time querying and navigating this database. In addition, Purdue researchers are working on parsing text from tweets and police reports. Data from Bureau of Motor Vehicles records and exchange of messages among police officers could be included in the future.

Nontechnical Deployment Concerns – This paper is focused on technical questions, but there are substantial nontechnical issues around deploying such a system. Some countries have deployed digital surveillance systems that are totalitarian. Some communities face overpolicing. Although there are some technical approaches that could possibly limit abuse, such as recent work in fairness in machine learning models, the challenges are quite broad

[8] ↑https://www.postgresql.org/

[9] ↑https://www.citusdata.com/

and unlikely to be solved solely by technical measures. It is not possible to address them adequately in a short paper, nor solely from a computing perspective. Obtaining the efficiency benefits of systems like this one, while limiting the potential for abuse, is a broad challenge for both the field and society overall.

Organization of this paper – We cover related work in Section 2.2.2, then discuss basic SurvQ architecture in Section 2.2.3. We describe the user's workflow in Section 2.2.4 and provide some initial experimental results in Section 2.2.5.

2.2.2 Related Work

Querying over video is a substantial research problem that draws on work in several areas of computer science.

Although queryable video monitoring systems have existed for some time, the neural network revolution in image processing has changed many of the system opportunities as well as research challenges. Earlier systems were limited to recognizing simpler objects, such as license plates or faces [23, 24]. One line of work assumes that video frames will be processed by a convolutional neural network (CNN) and is primarily concerned with optimizing their execution. NoScope [25] offers several optimizations, including training of inexpensive proxy models and selective frame differencing. The Tahoma [26] system creates multiple physical representations of videos, combined with creation of proxy models, in order to choose the most efficient one at query time. BlazeIt [12] offers optimizations for aggregation and limit queries that again rely on proxy model training. Focus [27] achieves runtime gains with a combination of inexpensive but low-quality CNNs to build an approximate query index, plus expensive high-quality CNNs after using the prebuilt index.

Many of these systems assume the existence of CNN-training capacity that would not be reasonable in our police use case. Their applications also focus heavily on traffic video use cases (whether fixed-camera or car-mounted) where they must detect (1) many examples of (2) common and (3) visually clear phenomena for downstream use by (4) analytical pipelines. In contrast, we are concerned with a human who often needs (1) a single example of (2) rare and (3) visually obscure phenomena for downstream use by (4) a human-intensive investigation. As a result, SurvQ does not have to generate huge quantities of results; it can exploit the natural duplication of imagery common in video in order to find a small number of difficult but high-value query results.

Other video query systems focus on traditional systems-centric optimization methods. SVQ (Streaming Video Queries) [28] is a system for running declarative SQL-style video stream queries, with a focus on counts and spatial constraints on objects in a frame. It optimizes query execution by applying a set of inexpensive filters (such as object counts) on video frames before running expensive object detection algorithms, thereby eliminating frames that have low potential of being a match for the query. Optasia [29] is a large-scale relational video query processing system that focuses on surveillance camera data. Its optimizations mainly focus on deduplicating work and choosing chunks for parallelization. Some of its approaches might be useful for SurvQ. VideoStorm [30] is an analytics system that uses a compute cluster to process thousands of concurrent analytical queries. It is primarily concerned with scheduling and resource allocation questions around those queries.

Some researchers have focused on detecting a larger set of items in video imagery. The Panorama system [31] represents the problem of unknown objects as an unbounded vocabulary problem. This system uses an ML-heavy approach that asks users to manually label unknown objects before automatically retraining novel image classifiers. This is an interesting human annotation problem, but is both computationally heavyweight and likely unnecessary in a concrete application with a relatively fixed set of objects. Techniques such as zero-shot [32] and one-shot [33] learning have been proposed for supporting new object properties, but they require significant manual intervention for model retraining as well as the provision of metadata and/or more labels, *i.e.*, identifying more object properties. As video surveillance applications usually have time-starved and non-technical users, these techniques are difficult to implement.

All of the above systems focus on the data system, without extensive attention paid to the human in the visual analytical loop.

2.2.3 Architecture

The architecture of SurvQ is divided into *ingestion* and *retrieval* systems. These two systems may be operating in parallel.

2.2.3.a Data Ingestion

Figure 2.8 shows the initial data ingress step. SurvQ consumes video feeds in real time or retrospectively. When video data arrives at the Video Server, SurvQ archives it in storage and then applies a a pipeline of processing steps:

1. Video is converted to MP-4 (if it is not already captured in this format) and down-sampled to one frame per second (there is no sense running property identification more often than this, and we may be able to run less frequently).

2. YOLO [2] is used to identify *people* objects in each frame. YOLO was chosen because it is very efficient at run time and has the best chance of keeping up with the large number of video streams. YOLO also has built-in detection for some of the properties from Table 2.1, for example bicycles.

3. Each YOLO-detected object is then further examined to discern its *object properties*, initially the features in Table 1.

2.2.3.b Granular-level Property Identification in Videos

Property identification is performed using three different approaches:

- **Color analysis.** The combination of YOLO class and some color analysis can yield a simple but effective property detector. We have segmented all YOLO-detected person objects into sections (e.g. lower half, upper 10% etc.). If the dominant color of the lower half is blue then chances are the person is wearing jeans. Between shape analysis, color analysis and common sense reasoning, we can detect about half of the objects in Table 2.1.

- **Traditional deep learning.** Purdue is applying traditional deep learning to the object property identification problem. This requires tagging imagery from West Lafayette with labels to construct training data. We have found that a suitable detector requires O(1000) images to be successful. Student labor is being used for this substantial task.

- **Transfer learning.** Purdue trained a CNN-based attribute detection classifier on the PA-100K dataset [34]. PA-100K dataset contains 100,000 pedestrian images from real outdoor surveillance cameras annotated with 26 attributes. We use YOLO as our backend before the frames are passed to the pretrained classifier. We created a mapping between 26 features from PA100K dataset and 31 features of interest in Table 2.1 based on whether they are visually synonymous: for example, *short sleeves* in the PA-100K dataset is mapped to *T-shirt* in our taxonomy. All of the features from PA100K are similar to some properties of interest to us, and we could find an exact mapping for more than one-third of the properties.

Currently, all three classifiers are being actively improved. However, a few results are already apparent. First, color/shape analysis works well and has the great advantage that training data is not required. Second, transfer learning is proving difficult, because of the variance in quality of frames between training data set and surveillance video. This same issue was a problem in transfer learning to identify jeans. Our experience is that data derived from dissimilar video is not useful in helping to solve our problem. We are hopeful, however, that our classifiers trained on West Lafayette video will work successfully on New Hampshire video. Lastly, training a novel deep learning system is a challenge because of the amount of training data required. Without a ready data set of tagged images, generating training data is a very expensive proposition. Without student labor, the cost could well be prohibitive.

Active Learning and Color Management – Active learning is well understood in a deep learning context. However, of particular interest to us is feedback to our color algorithms for automatic improvement. Colors perceived by the camera can be influenced by the time of day, weather, and other physical properties in the scene so that simply detecting "blue" on a person figure requires some adjustment. To perform color analysis in a segmented area, the RGB value at each pixel is retrieved. The set of RGB values for the standard colors (red, green, yellow, *etc.*) is defined in a reference color map. To infer the color of a pixel, we calculate the color distance to each standard color and choose the closest one [10].

The color for any region of interest was then found by majority vote. To incorporate active learning, we plan to test moving the reference colors in color space based on user feedback.

Finally, the output of the classifiers is stored in Postgres along with suitable metadata, and pointers to the archived raw video.

2.2.3.c Data Retrieval with Incomplete Data and Meeting Information Needs Over Time

The right-hand side of Figure 2.8 shows the simple architecture we use for querying video. At run time, our system expects a user query, most likely derived from information in a police incident report, such as the West Lafayette document seen in Figure 2.9. (We also have code that parses the actual incident reports to extract the description of the incident.) The user's query is converted to SQL and defined as a trigger to the Postgres DBMS. In this way, a historical query is run to find the data of interest in the past, and a Postgres trigger will find data of interest as it is loaded into the DBMS in the future. Section 2.2.4 describes in detail the query and interaction cycle from the user's perspective.

We currently ingest parsed tweets into the database. In the future, we expect to search both video and tweets for the properties of interest in Table 2.1. Note that the scope of a trigger system can be multiple tables, so joining multiple data sources is straightforward.

2.2.4 User Workflow: The Police Detective In The Loop

Police detectives receive information about events they need to investigate in the form of an incident report, like that of Figure 2.9. These reports contain information about the event that occurred, often including details of any suspects involved. We have two interfaces to obtain data about an incident. The first is a form-based UI shown in Figure

[10] ↑https://www.compuphase.com/cmetric.htm

2.10. With it, detectives can input the incident details easily. The form has fields to filter on time, location and suspect characteristics. The results are translated into SQL queries and triggers, allowing non-programmers to easily interact with the system. The second system automatically parses West Lafayette incident reports to obtain required information.

An example of the analytical interaction steps a detective may take with SurvQ is shown in Figure 2.11. In (1) the user will visit the creation page for the incident and enter the appropriate details. Upon submission, the user is redirected to an investigation page that can be revisited at any time. The investigation page contains all the details relevant for the incident. This includes event information, processing progression and matching video clips. In (2) there are three possible *viewpoints* the user can choose:

- **List:** Displays all returned results compactly so the user can quickly view all matches.
- **Map:** Aggregates video that occurred in close proximity and displays the resulting clusters on a map
- **Timeline:** Aggregates video that occurred close in time and displays the bucketed groups in order

These three different viewpoints are shown in Figures 2.12 and 2.13. In step (3), the investigator will search through the returned video. When an investigator finds a video useful, they mark it as important. Each viewpoint can apply a filter to display only marked video. In step (4), after they've gone through the video, they can select a different viewpoint to make additional passes over the video data. This search-and-mark process comprises much of the detective's analytical work. In the future, we believe that moving cameras (mounted, say, on a police car) may be a potential source of novel video data, and will pose new interaction challenges for investigators attempting to find suspects in the video database.

Additional video may enter the system after initial creation and investigation. Investigators following an incident in real time can use SurvQ to subscribe to long-running queries and thereby receive alerts about new data. Postgres triggers are made on incident creation to check for property matches. Users can see new notifications in the investigation view and home page for easy viewing and access.

2.2.5 Runtime Performance: Scalability of Data Ingestion and Retrieval

We have run our ingest system and our query system in parallel on real and simulated queries and data. To assess performance, we consider SurvQ as three components: video ingest, YOLO processing, and database activity. Our goal was to make SurvQ performant enough to handle an urban camera deployment, including the West Lafayette use case.

We maintain a collection of web servers to handle video upload. Upsteam processing must generate video in 1-minute MP-4 files. Future work would be required to handle other video formats. A low cost web server can support 4-5 concurrent video feeds.

YOLO processing runs well on a GPU-equipped server. A single server can perform object recognition and color analysis in approximately 15 seconds per 1 minute of video. Thus, each YOLO processing server can handle around 4 incoming feeds.

Our major performance concern was how our use of trigger functions in Postgres would scale. In SurvQ, a trigger function for each incident is run every time YOLO results are inserted. We run YOLO on video at a rate of 1 frame per second, or 60 frames per minute. West Lafayette surveillance data averages 5 persons per frame. Since the West Lafayette use case has at most 200 video sources, we would expect our system to receive around 60 * 5 * 200 = 60000 inserts per minute. We need to be able to handle these insertions in under 60 seconds worth of time. Figure 2.14 shows that trigger invocation can easily keep up.

As such, the dominant cost for West Lafayette is the number of YOLO servers to process the incoming video load. Given their computing budget, it is not cost effective to perform ingest-time property identification on all video. Instead we have implemented resource-available classification via a priority system. In this way, highest priority feeds are classified at ingest time, and deferred processing is performed when necessary. Of course, when processing is deferred, it is performed at query time. Hence, in the worst case, only video relevant (in space and time) to an incident is classified. Our current system assigns a priority to a video feed equal to the number of incidents it matches in space and time. We expect to investigate more complex schemes in the future.

In our opinion, deferred processing is more reasonable than trying to do multi-step identification, as in BlazeIt [12]. The BlazeIt sampling approach can be effective in counting settings but these do not apply to our surveillance analyst use case; moreover, the BlazeIt approach requires assumptions about the detected objects that may not apply for our use case, as well as additional model training overhead. A better way to perform multi-step identification for the surveillance analyst use case is to first do coarse temporal sampling, e.g. down sample to 6 frames a minute from 60 frames per minute. Then perform finer granularity sampling at query time.

Table 2.1 indicates running and walking as properties of interest. To accomplish this, we perform inter-frame analysis. It is straightforward to calculate geographic displacement of objects with the same color properties. With fixed cameras this is working well, and we plan to expand to cover moving cameras. Also, we expect to continue this thrust to explore social behavior of suspects and predict their intentions, for example, a man with a black hood is wandering back and forth in front of a bank.

We are also looking for techniques to identify rare events, such as tattoos. Deep learning is difficult in this case because of the absence of training data. Even color analysis is difficult because of the dearth of examples to test algorithm design on.

2.2.6 Conclusion and Future Work

In this paper, we have introduced SurvQ, a human-in-the-loop system for analyzing surveillance video. We have described a database backend that can scale to practical video volumes, as well as an interface that dramatically lowers the human costs of video-driven investigations. Although our experimental results are preliminary, the performance numbers are already promising, and our intended user base of law enforcement officers have expressed extreme enthusiasm for the software artifact. In the future, we plan to grow the video datasets under management and extend the investigation interface to include mobile cameras, both vehicle- and body-mounted. We also plan to apply the system to novel investigation scenarios such as management of warehouses or construction sites. Finally, we will continue to investigate "minimal user surface" ML deployments in a video context.

2.3 Human Attribute Recognition from Unstructured Text (HART)

Specifically, we explore the problem of identifying properties describing humans from unstructured text. As discussed in SurvQ [35], a finite number of object properties such as, GENDER, RACE, BUILD, HEIGHT, CLOTHES, etc. are used in profiling a person-of-interest to search for them. We denote object-properties used in person profiling as $\mathcal{O}_H$.

(2.1) The sentence "a **white male** with **medium** build was seen in Vernon St., *wearing* **white jeans** and **blue shirt**" describes object-properties of a †person:

1. GENDER = male,
2. RACE = white,
3. BUILD = medium,
4. *CLOTHES = {jeans, shirt},
5. UPPER-WEAR-COLOR= {white},
6. BOTTOM-WEAR-COLOR = {blue}, and
7. RELATION = {wearing, †Person, *Clothes}.

2.3.1 Problem Definition

Definition 2.3.1 (Wordnet Synsets). *Wordnet[36] is a lexical knowledge base where words are organized in a hypernym tree based on their origin. Words are grouped into Synsets based on their synonyms. Wu-Palmer distance calculates the similarity between word meanings based on how similar the word senses are and where the Synsets occur relative to each other in the hypernym tree. Given the synsets of two strings s_{t_1} and s_{t_2}, and the LCS (Least Common Subsumer) between them, the Wu-Palmer distance is:*

$$wpdist(s_{t_1}, s_{t_2}) = 2 * \frac{depth(lcs(s_{t_1}, s_{t_2}))}{depth(s_{t_1}) + depth(s_{t_2})} \tag{2.2}$$

Definition 2.3.2 (Natural Language Inference). *Given a hypothesis h and a premise p, Natural language inference (NLI) is the task of determining the probability Pr of the hypoth-*

esis being true (entailment E), false (contradiction C) or undetermined (neutral N). NLI determines the best label l:

$$\arg \max_{l \in \{E,C,N\}} Pr(l \mid h, p)$$

Problem 2.1 (Human Attribute Recognition from Text)**.** *Given a large text T with T_s sentences, each with $|w|$ tokens, the problem of human attribute recognition from T is to*

1. *identify the set of sentences $C_s \subset T_s$ that describes properties of a person,*
2. *expose the set of object-properties $\mathcal{O}_H$ from C_s and*
3. *extract the set of values z_p of the identified properties $\mathbf{o}_p$.*

Our problem setting assumes that the set of key-phrases (Q_H) often used in sentences describing properties of a person are either known (provided by domain experts), or a small amount of annotated documents are provided to identify Q_H manually. In Example 2.1, $Q_H = \{\texttt{wearing}\}$. The first assumption is derived from literature in pedestrian attribute recognition from visual and textual modalities, and the second assumption holds as small amount of curated data is always available for a problem setting. Note that, $(Q_H \cap \mathcal{O}_H) \neq \{\phi\}$. Candidate sentences are sentences in the text that mentions phrases similar to the key-phrases within an empirical threshold value.

Definition 2.3.3 (Candidate Sentences)**.** *Given a collection of sentences T_s, key-phrase for describing an object in text $q_H \subset Q_H$, and an empirical threshold θ_H, Candidate sentence is*

$$C_s = \{s : s \in T_s, q_H \in Q_H \;\mid\; SIM(q_H, s) > \theta_H\} \tag{2.3}$$

2.3.2 Methodology

We now describe the property identification technique for unstructured texts to extract *attribute-based properties* from large text documents. Our algorithm considers the full document as input and reports a *collection* of object-properties and their set of values, as output. To this end, we first identify the candidate sentences C_s from a collection of sentences T_s by searching for the key-phrases (q_H) using pre-trained language representation models and lexical knowledge bases. Then, we propose individual property-focused models to extract the attributes and their corresponding values using the syntactic characteristics (i.e., parts-of-speech) and lexical meanings of the tokens in the *Candidate Sentences*. Our heuristic search algorithm, POSID iteratively checks the tokens in the candidate sentences and based on the assigned tags in accordance with their syntactic functions identifies the properties in $\mathcal{O}_H$ and their values.

2.3.2.a Candidate Sentence Extraction

A naive approach to this task would be to consider it as a supervised classification problem given enough training data. Since during this work, the primary goal was to define on-demand models that works in absence of training data, we designed this as a similarity search problem using pre-trained and lexical features, where the similarity between sentence and key-phrase needs to reach an empirical threshold. We now proceed to describe the different methods used to identify C_s.

(i) Pattern Matching.

As a baseline heuristic model, we implemented the **Regular Expression (RE)** Search on T_s. Since we consider all sentences in the document as input corpus, if it describes multiple persons, this model captures all of the sentences describing a person as C_s. Individual mentions are differentiated in later stages. For RE, $SIM(q_H, s) \in \{0, 1\}$. Given the key-phrase q_H, the RE pattern searches for any sentence mentioning it:

$$\texttt{[\^{}]*}q_H\texttt{[\^{}.]+}$$

(ii) Similarity using Tokens

Similarity between q_H and s is calculated based on the similarities between tokens $w \in s$ and q_H. A single model is used to embed both w and q_H into the same space. We used two different token representation models.

$$SIM(q_H, s) = \max_{w \in s} SIM(q_H, w) \tag{2.4}$$

(a) ***Word Embedding.*** Tokens in each sentence and in the key-phrase are represented by **Word2Vec** [37] embeddings. If there are multiple tokens in a key-phrase, the average of the embeddings are used. We use cosine similarity as the distance metric. Given u_{q_H} and u_w are the final embedding vectors for q and w,

$$SIM(q_H, w) = cos(u_{q_H}, u_w) = \frac{u_{q_H} \cdot u_w}{\|u_{q_H}\| \cdot \|u_w\|} \tag{2.5}$$

(b) ***Word Synsets.*** Tokens and key-phrases are represented by **Wordnet** [36] synsets in `NOUN` form. For similarity/distance metric, we used the Wu-Palmer similarity [38]. Given the synsets of q and w are s_{q_H} and s_w,

$$SIM(q_H, w) = wpdist(s_{q_H}, s_w) \tag{2.6}$$

(iii) Classification Model

The similarity search problem is redesigned as a classification problem where the sentences are considered as input sequence, and the key-phrases are considered as labels. Probability of sequence s belonging to a class q_H is then considered as the similarity between a sentence and a key-phrase. To that end, following Yin et al. [39], we used pre-trained natural language inference (NLI) models as a ready-made zero-shot sequence classifier. The input sequences are considered as the NLI premise and a hypothesis is constructed from each key-phrase. For example, if a key-phrase is `clothes`, we construct a hypothesis *"This text is about clothes"*. The probabilities for *entailment* and *contradiction* are then converted to class label probabilities. Then, both the sequence and the hypothesis containing the class label are encoded using a sentence level encoder Sentence-BERT [40] (**SBert**). Finally, we use the NLI model to calculate the probability P. Given SBERT embedding of a sequence s is denoted with B_s,

$$SIM(q_H, s) = P(s \text{ is about } q_H \mid B_s, B_{q_H}) \tag{2.7}$$

(iv) Stacked Models

While **RE** search relies on specific patterns and returns exact matches, the other models calculate a soft similarity, $0 \leq SIM(q_H, s) \leq 1$. Hence if initial results from **RE** search returns no result for all the key-phrases we use **Wordnet** or **SBert** model to identify semantically similar sentences to the key-phrases.

2.3.2.b Iterative Search for Properties

We now formally describe the POSID algorithm, which uses the models described in Section 2.3.2.a. We start with the observations that led to the POSID algorithm.

Observations. We make the following observations:

(O1) Common to **O4.2.4**, object-properties have the single and multiple value contrasts.

(O2) Some properties follows specific patterns such as, GENDER = `{male, female, man, woman}`, whereas some properties have variable values, as shown in **O4.2.5**.

(O3) Adjectives (ADJ) are used for naming or describing characteristics of a property, or used with a NOUN phrase to modify and describe it.

(O4) Property values can span multiple tokens, but they tend to be consecutive.

(O5) Property values for CLOTHES generally include the color, a range of colors, or a description of material.

(O6) CLOTHES usually is described after consecutive tokens with VERB tags V_{DG} (such as, gerund or present participle (VBG), past tense (VBD) etc). If proper syntax is followed, an entity is described with a VBD followed by a VBG. In most cases, mentioning `wearing`.

(O7) After a token with VBG tag, until any ADJ or NOUN tag is encountered, any tokens describing a P_{DCP} {Determiner, Conjunction, Preposition}, or a Participle, or Adverb is part of the property-name. An exception would be any P_{PAV} {participle, adverb, or verb} preceded by any $P_{P\epsilon}$ {pronouns or non-tagged tokens}, which ends the mention of a property-name.

1 Algorithm 1: POSID(T_s, $\langle o_p, z_p \rangle$)

```
$fo_p \leftarrow \{$GENDER, RACE, HEIGHT$\}$
$COLOR_{syn} \leftarrow \mathsf{SYNSETS}(\text{``}color\text{''}, Noun)[0]$
$C_s \leftarrow \mathsf{extractCSRegex}(T_s, Q_H)$
if $C_s == \phi$ then
    $C_s \leftarrow \mathsf{extractCSModel}(T_s, Q_H)$
fi
$\mathcal{O}_H \leftarrow \emptyset$   // Collection of $\langle o_p, z_p \rangle \equiv \langle name, values \rangle$ pairs
for $s \in C_s$ do
    for $o \in fo_p$ do
        // $L_z$ is last token in $s$ which is a property-value
        $L_z = \mathsf{regexPROP}(s, \mathbf{o}_p)$
        $\mathcal{O}_H.\mathsf{APPEND}(\mathbf{o}_p, L_z)$
    endfor
    $s_p = \mathsf{regexPROP}(s, \text{CLOTHES})$
    if $s_p == \phi$ then $s_p \leftarrow s \setminus L_z$
    fi
    $N_{\text{idx}} \leftarrow \emptyset$   // Index-List for property-name
    $D \leftarrow \emptyset$   // List for property-values
    $T_o \leftarrow \mathsf{TOKENIZE}(s_p)$   // List of tokens from $s_p$
    $T_a \leftarrow \mathsf{POS}(T_o)$   // List of $\langle$token, POS-tag$\rangle$ from tokens
    $checkPOS(T_a, N_{\text{idx}}, D, d_{color}, S_w)$
endfor
return $\mathcal{O}_H$
```

2 Algorithm 2: checkPOS($T_a, N_{\mathrm{i}dx}, D, d_{color}, S_w$)

```
for $(w, t) \in T_a$ do
    // $w_{\mathrm{i}}$ and $t_{\mathrm{i}}$ is token and POS-tag at $\mathrm{i}^{th}$ index in $T_a$
    if $t_1$ is $VBD$ then continue
    if $t_2$ is $VBG$ and $t_1$ is $VBD$ then continue
    if $t_{\mathrm{i}} \in P_{DCP} \cup P_{PAV}$ then
        if $t_{\mathrm{i}} \in P_{PAV}$ and $t_{\mathrm{i}-1} \in P_{P_\epsilon}$ then break
        $N_{\mathrm{i}dx}$.APPEND(i)
    elseif $t_{\mathrm{i}}$ is $ADJ$ then
        $N_{\mathrm{i}dx} \leftarrow \emptyset$  // re-initialize name index-list
        $D$.APPEND($w_{\mathrm{i}}$)
    elseif $t_{\mathrm{i}}$ is $NOUN$ then
        $S_w \leftarrow$ SYNSETS($w_{\mathrm{i}}, NOUN$)
        $N_{\mathrm{i}dx}, D, d_{color}$ = matchCOLOR($S_w, N_{\mathrm{i}dx}, D$)
        if $d_{color}$ then continue
        $N \leftarrow w_{\mathrm{i}}$
        $N \leftarrow$ writePROP($N_{\mathrm{i}dx}, N, T_a$)
        // finalize property-name & assign the values
        if $t_{\mathrm{i}-1}$ is $NOUN$ and
            $\mathcal{O}_H[-1].name == w_{\mathrm{i}-1}$ then
                CONCAT($\mathcal{O}_H[-1].name, w_{\mathrm{i}}$, “”)
        else $\mathcal{O}_H$.APPEND([$N, D$])
        $N_{\mathrm{i}dx} \leftarrow \emptyset, D \leftarrow \emptyset$  // re-initialize Lists
        else break
endfor
```

Algorithm 1 presents the pseudocode of the search technique POSID, which takes the sentences in a document T_s as input and returns the *collection* of object-properties and their set of values, $\langle\langle o_p, z_p\rangle\rangle$ as output. In case of an implicit mention of clothes, we made an assumption that description of CLOTHES are always followed by GENDER, RACE, and/or HEIGHT.

Lines 3 - 5 Extract the candidate sentences with the RE-SEARCH. If results are empty, extract them with semantic or classification models. Set of key-phrases Q_H is provided by the system.

Lines 9 - 12 Iteratively search for all the finite-valued properties {GENDER, RACE, HEIGHT} in each C_s and append them to output.

`regexPROP` is a regular expression matching function that takes sentence s and property-name o_p as input, and outputs

1. property-value z_p, if o_p is a finite-valued property, or

2. partial sentence s_p, if o_p is a variable-valued property.

Each o_p is mapped to a search-string pattern, s_R in T.

Lines 14 - 15 For CLOTHES, `regexPROP` returns either a partial sentence s_p starting with `wearing`, or an empty string. In case of an empty string, extract the remaining string from L_z after discarding the extracted values in lines 9 - 12.

Lines 26 - 27 If first and second token is verb, it is the start for the RELATION property. Following **(O6)**, ignore consecutive verbs until another tag is encountered.

Lines 28 - 30 Following **(O7)**, capture tokens from a VERB until any pronoun or non-tag as free-form property value for CLOTHES.

Lines 31 - 33 Capture the adjectives as clothes descriptions, and initialize the next property.

Lines 35-37 For noun descriptors in the value i.e., grey **dress** pants, compare the wordnet-synset meaning for `color` ($COLOR_{syn}$) to the noun-token meaning. Since a description is encountered, name-index is re-initialized for the next property-name.

Lines 38-39 If a noun-phrase is not a color, it is considered as cloth-name with multiple tokens i.e., *dress pants, tank top, dark clothing.* Populate the property-name by backtracking the name-index list.

Line 44 If previous token is NOUN and does not match last token of the previous property-name, description for next property has started. Finalize the current property name and value by appending it to result. Otherwise, in line 43, amend the last inserted property-name by appending the current token to it.

Generalization.

Algorithm 1 assumes that the property identifier is intended for human-properties. POSID can be generalized to any object-properties in text as long as the property-names and type of values are known. Search-string for fixed-valued properties have to be re-designed. Variable-valued properties following some degree of grammatical structure, would be covered by the iterative search pattern in POSID. COLOR will be replaced by the phrase that describes the properties in the corresponding system. Q_H are highly non-restrictive phrases and can be constructed from entity types or entity names.

2.3.3 Experiments

2.3.3.a Dataset Construction.

For property identifiers in textual modalities, we build a collection of text data, named **InciText** dataset from newspaper articles, incident reports, press releases, and officer narratives from the local police department. We scraped local university newspaper articles to search for articles with keywords i.e., *investigation, suspect, 'person of interest'* and *'tip line phone number'*. InciText provides ground-truth annotations for 12 properties describing human attributes with most common being – GENDER, RACE, HEIGHT, CLOTHES and CLOTH DESCRIPTIONS (**COLORS**). Each report, narrative, and press release describes zero, one, or more persons.

2.3.3.b Settings.

For Word2Vec, we used the 300 dimensional pretrained model from NLTK [41] trained on Google News Dataset[11]. We pruned the model to include the most common words (44K words). From NLTK, we used the built-in tokenizers and the Wordnet package for retrieving

[11] ↑GoogleNews-vectors-negative300

the synsets and wu-palmer similarity score. For SBERT implementation, we used the zero-shot classification pipeline[12] from transformers package using the SBERT model fine-tuned on Multi-NLI [42] task. For part-of-speech tagging, we used the averaged perceptron[13] tagger model. The manual narratives in the InciTextdataset were excluded for property identification task. Query phrases used for C_s identification are: $q_H = \{$ `clothes, wear, suspect, shirts, pants` $\}$.

2.3.3.c Results

We compared the baseline RE-model with the other approaches in Section 2.3.2.a for finding C_s. Two different set of metrics were used for the evaluation of CLOTHES identification. (**Attr-only**) evaluates how efficiently the model identified all clothes, and (**Attr-value**) calculates the performance of the model in identifying both the attribute and its descriptive values. For Attr-value, a true positive occurs only when a valid *clothes* name and a correct description of that cloth is discovered. Figure 2.2 describes the performance of different candidate sentence extraction models based on the performance of CLOTHES identification. For the baseline, the group of tokens around `wear` returned three times better F1-score than any other q_H. With the other models, $q_H = \{$`clothes`$\}$ produced the best score. (RE + SBERT) stacked model performs best with 87% and 90% F1-Scores, for both metrics. Although (RE + Wordnet) has a higher precision score of 93% for Attr-only, it has a low recall score of only 65%, indicating over-fitting. Based on a property-frequency analysis, we showed the identification results for a subset of properties in $\mathcal{O}_H$ for InciText. Figure 2.3 shows the performance of POSID with (RE+SBERT) for stacked model (lines 3 - 5). For gender and race, the model showed the efficacy of the chosen search-pattern with 94% precision score. A recall score of 73% shows that most people follow similar style for describing gender and race. For *height* with only 57% recall score, a rule based model is not sufficient due to varied styling.

[12] ↑Zero-shot-classification

[13] ↑NLTK Perceptron

2.3.4 Conclusion and Future Works

Although human attribute recognition from videos and images has been well studied, we believe this is the first work that focuses on finding them from the text. [43, 44] used sentence encoders and dense neural networks to combine lexical and semantic features for finding similar sentences in electronic medical records and academic writing. We demonstrated the efficacy of HART, a human attribute recognition model from unstructured text, outperforming the baseline language models. In future, we would like to implement HART in other application domains and mission needs.

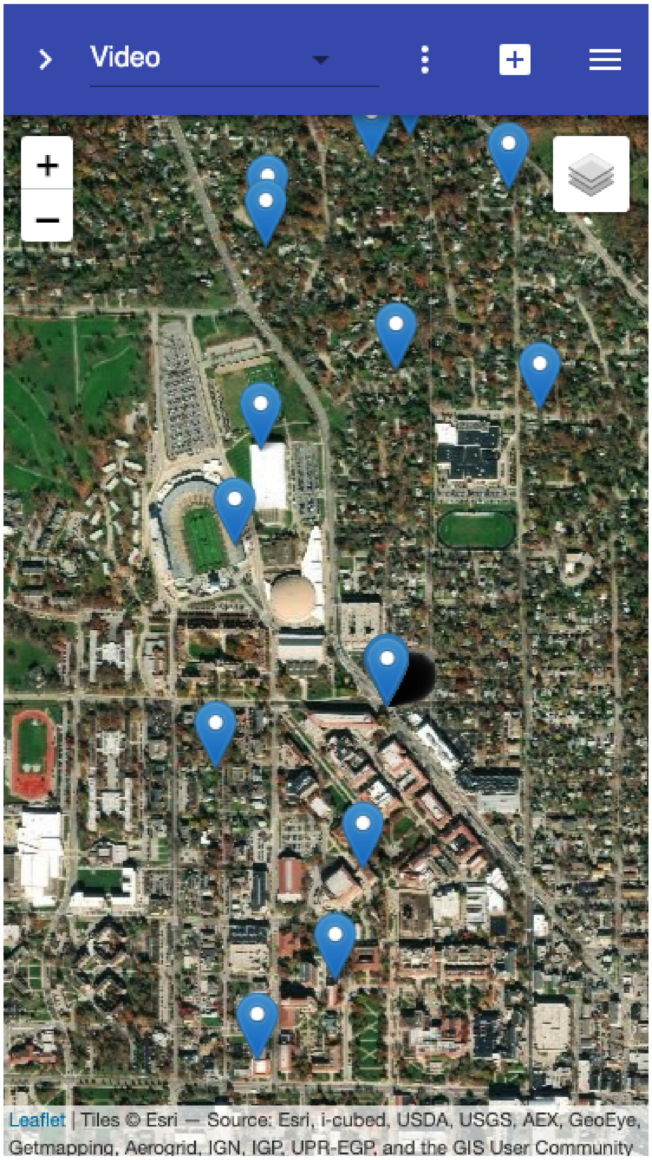

Video
+
−
Leaflet | Tiles © Esri — Source: Esri, i-cubed, USDA, USGS, AEX, GeoEye, Getmapping, Aerogrid, IGN, IGP, UPR-EGP, and the GIS User Community

Figure 2.6. Situational Knowledge on Demand proof-of-concept. Incoming streams of data shown in a Leaflet map.

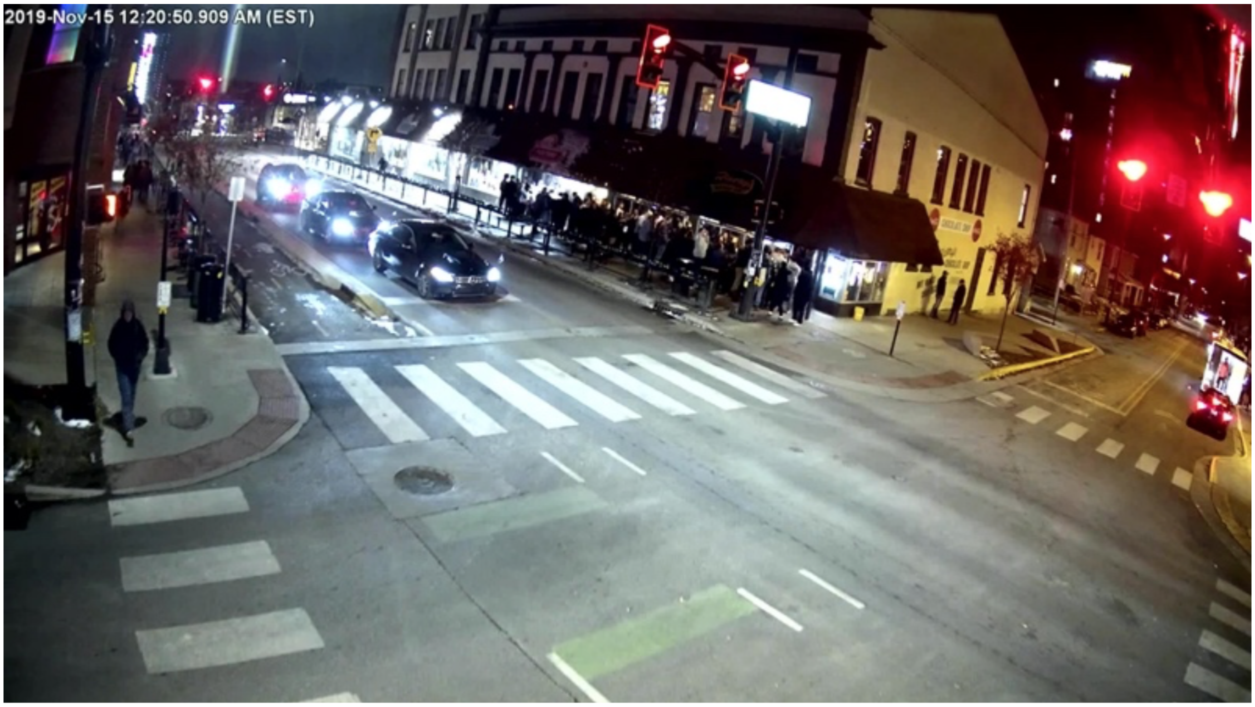


Figure 2.7. An example of surveillance footage from West Lafayette, IN.

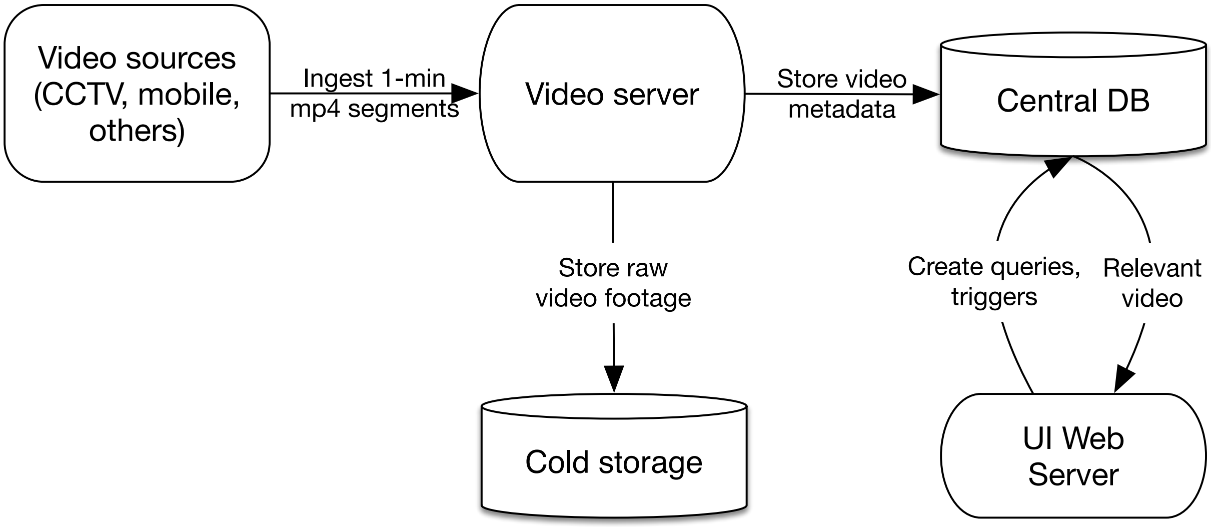
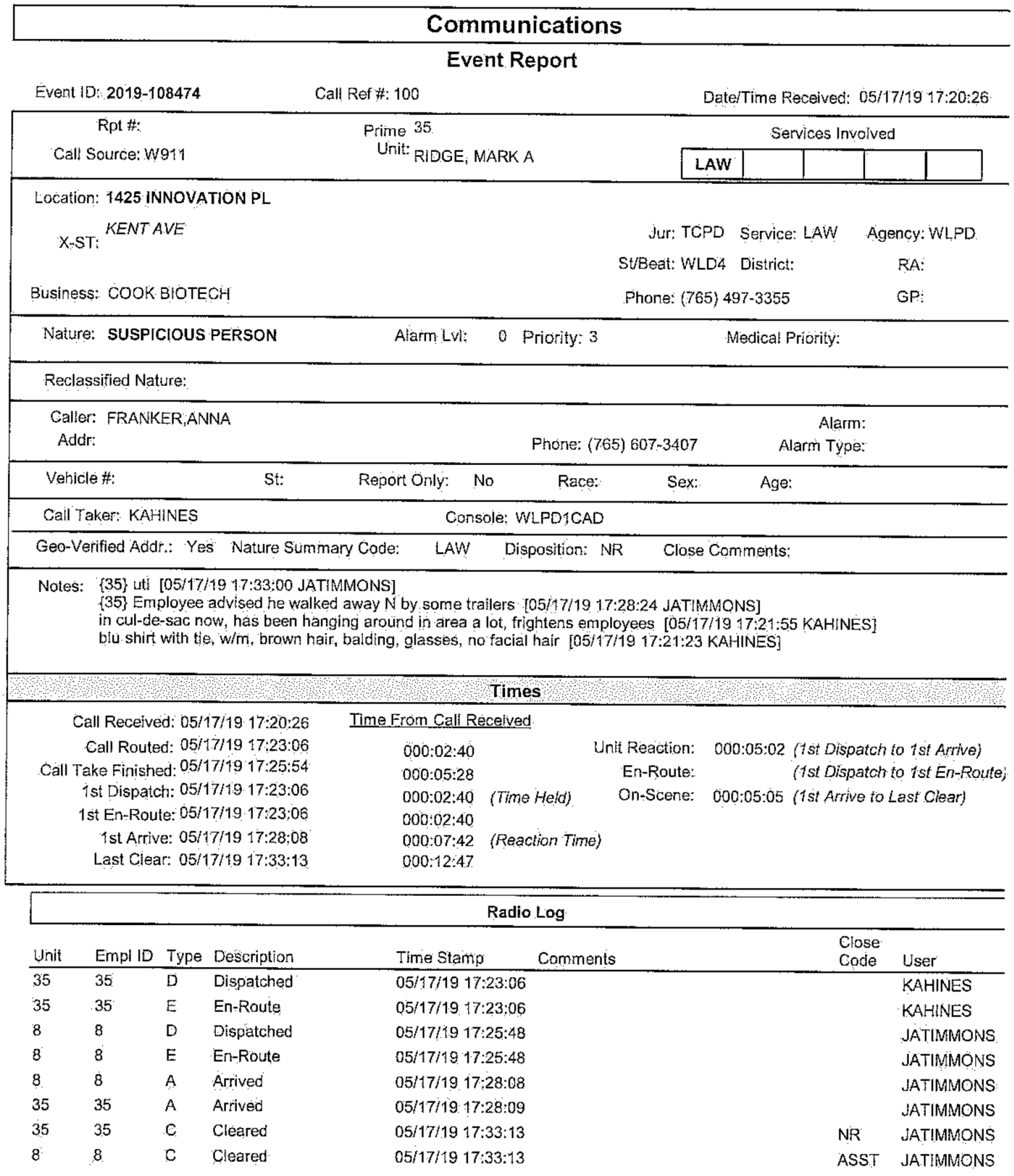


Figure 2.8. SurvQ video ingestion and retrieval architecture



Communications

Event Report

Event ID: **2019-108474** Call Ref #: 100 Date/Time Received: 05/17/19 17:20:26

Rpt #:
Call Source: W911

Prime Unit: 35
RIDGE, MARK A

Services Involved

LAW				

Location: **1425 INNOVATION PL**

X-ST: *KENT AVE*

Jur: TCPD Service: LAW Agency: WLPD
St/Beat: WLD4 District: RA:

Business: COOK BIOTECH Phone: (765) 497-3355 GP:

Nature: **SUSPICIOUS PERSON** Alarm Lvl: 0 Priority: 3 Medical Priority:

Reclassified Nature:

Caller: FRANKER,ANNA Alarm:
Addr: Phone: (765) 607-3407 Alarm Type:

Vehicle #: St: Report Only: No Race: Sex: Age:

Call Taker: KAHINES Console: WLPD1CAD

Geo-Verified Addr.: Yes Nature Summary Code: LAW Disposition: NR Close Comments:

Notes: {35} utl [05/17/19 17:33:00 JATIMMONS]
{35} Employee advised he walked away N by some trailers [05/17/19 17:28:24 JATIMMONS]
in cul-de-sac now, has been hanging around in area a lot, frightens employees [05/17/19 17:21:55 KAHINES]
blu shirt with tie, w/m, brown hair, balding, glasses, no facial hair [05/17/19 17:21:23 KAHINES]

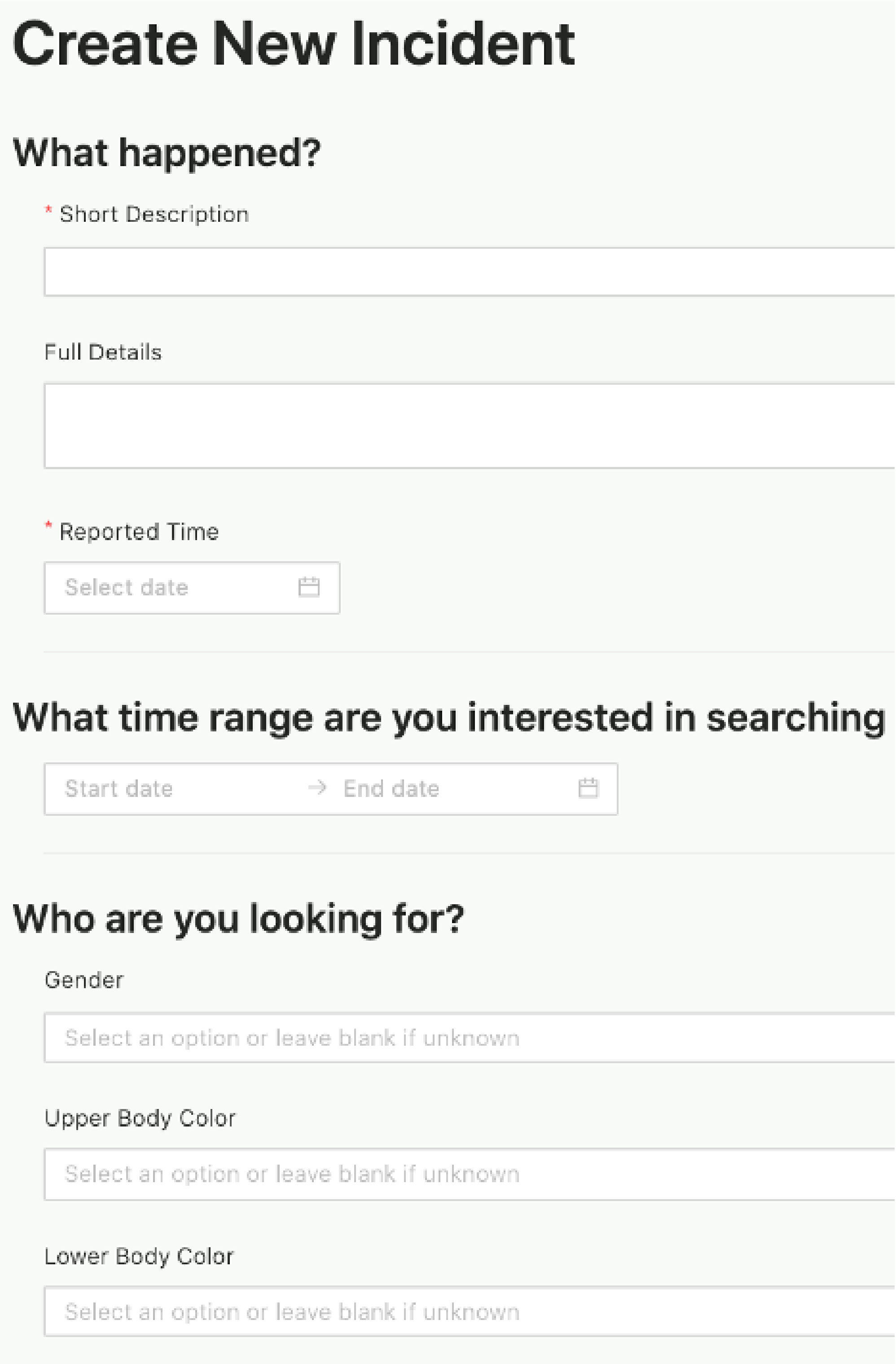
Times

		Time From Call Received				
Call Received:	05/17/19 17:20:26					
Call Routed:	05/17/19 17:23:06	000:02:40		Unit Reaction:	000:05:02	*(1st Dispatch to 1st Arrive)*
Call Take Finished:	05/17/19 17:25:54	000:05:28		En-Route:		*(1st Dispatch to 1st En-Route)*
1st Dispatch:	05/17/19 17:23:06	000:02:40	*(Time Held)*	On-Scene:	000:05:05	*(1st Arrive to Last Clear)*
1st En-Route:	05/17/19 17:23:06	000:02:40				
1st Arrive:	05/17/19 17:28:08	000:07:42	*(Reaction Time)*			
Last Clear:	05/17/19 17:33:13	000:12:47				

Radio Log

Unit	Empl ID	Type	Description	Time Stamp	Comments	Close Code	User
35	35	D	Dispatched	05/17/19 17:23:06			KAHINES
35	35	E	En-Route	05/17/19 17:23:06			KAHINES
8	8	D	Dispatched	05/17/19 17:25:48			JATIMMONS
8	8	E	En-Route	05/17/19 17:25:48			JATIMMONS
8	8	A	Arrived	05/17/19 17:28:08			JATIMMONS
35	35	A	Arrived	05/17/19 17:28:09			JATIMMONS
35	35	C	Cleared	05/17/19 17:33:13		NR	JATIMMONS
8	8	C	Cleared	05/17/19 17:33:13		ASST	JATIMMONS

Figure 2.9. The populated incident report



Create New Incident

What happened?

* Short Description

Full Details

* Reported Time

Select date

What time range are you interested in searching

Start date → End date

Who are you looking for?

Gender

Select an option or leave blank if unknown

Upper Body Color

Select an option or leave blank if unknown

Lower Body Color

Select an option or leave blank if unknown

Figure 2.10. The Incident Creation Form.

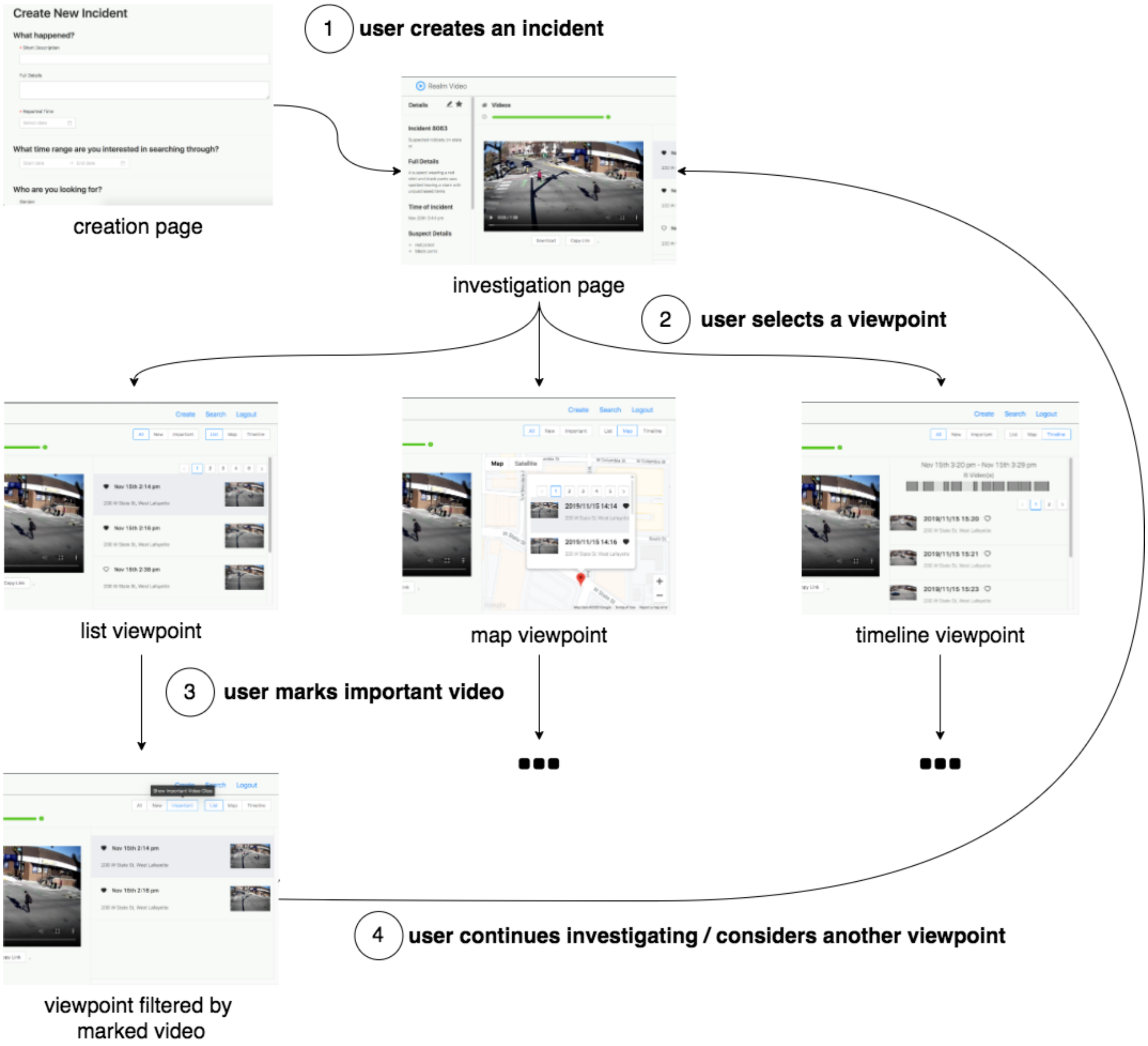
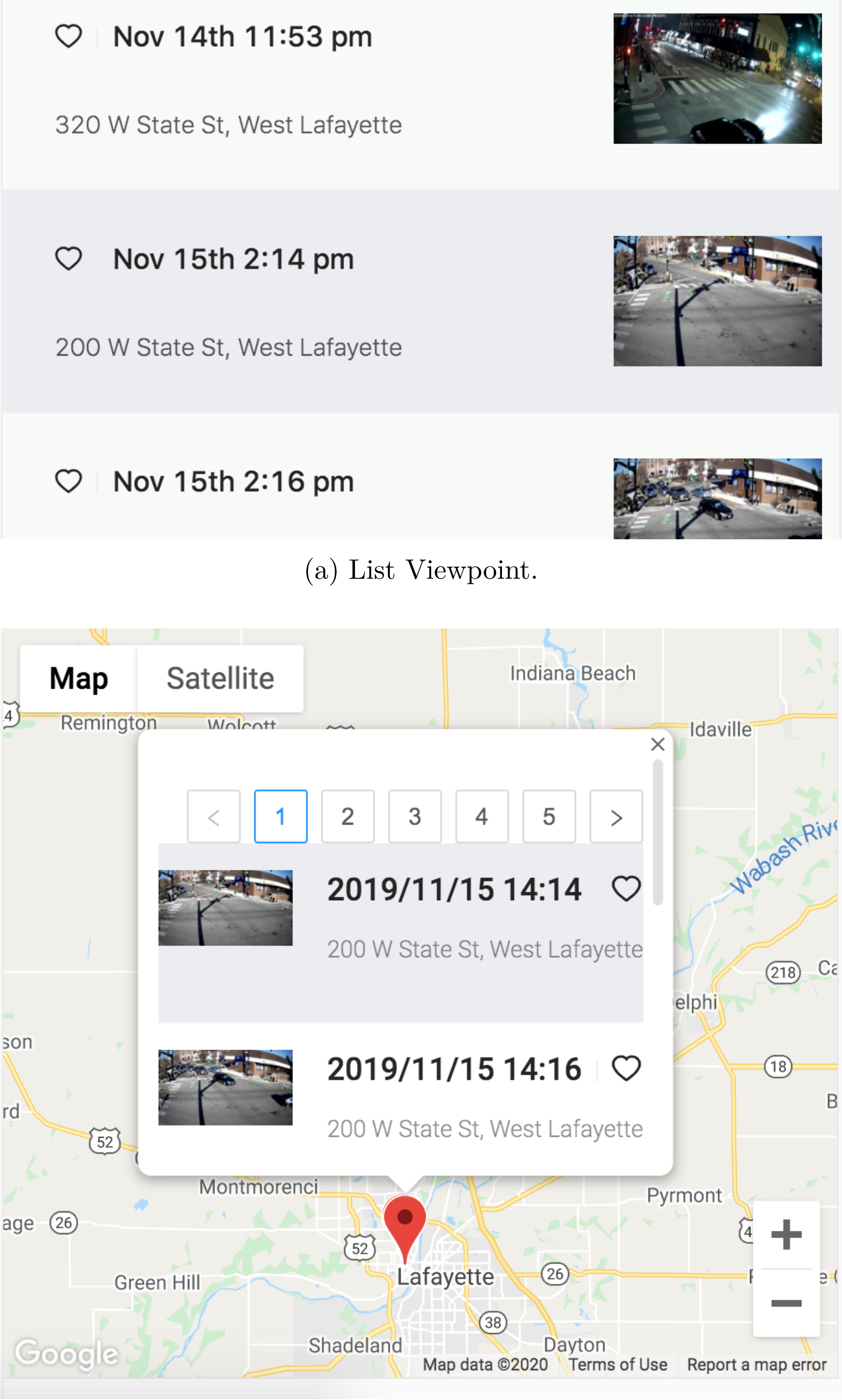


Figure 2.11. Example actions a user might take with SurvQ.



♡ Nov 14th 11:53 pm

320 W State St, West Lafayette

♡ Nov 15th 2:14 pm

200 W State St, West Lafayette

♡ Nov 15th 2:16 pm

(a) List Viewpoint.

Map | Satellite

< 1 2 3 4 5 >

2019/11/15 14:14 ♡

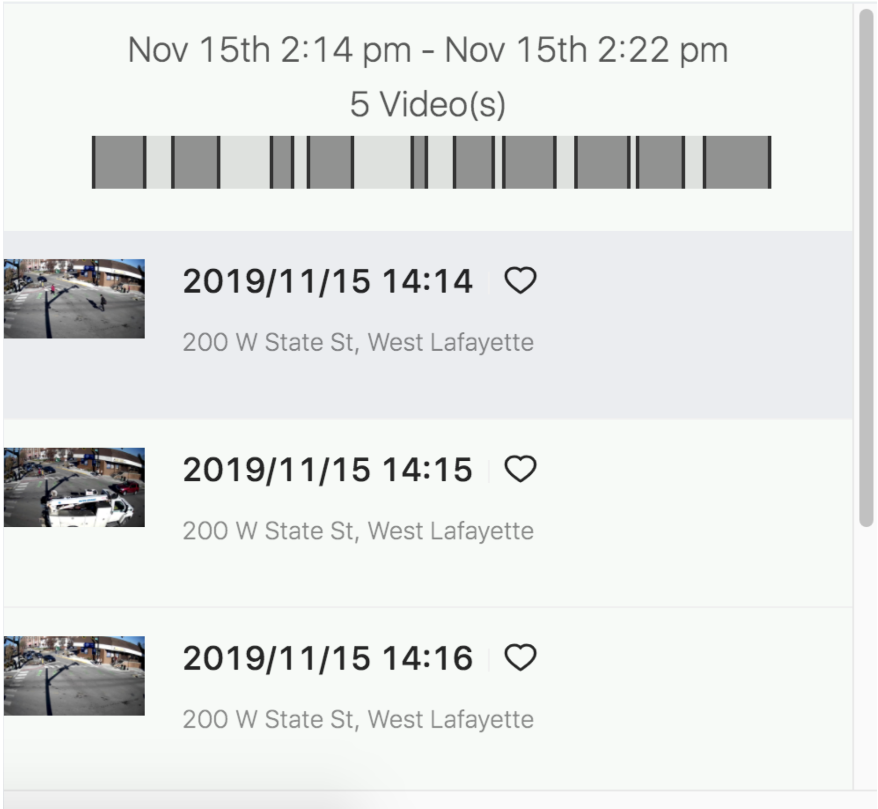
200 W State St, West Lafayette

2019/11/15 14:16 ♡

200 W State St, West Lafayette

(b) Map Viewpoint.

Figure 2.12. The *list* and *map* viewpoints.



Nov 15th 2:14 pm - Nov 15th 2:22 pm

5 Video(s)

2019/11/15 14:14 ♡

200 W State St, West Lafayette

2019/11/15 14:15 ♡

200 W State St, West Lafayette

2019/11/15 14:16 ♡

200 W State St, West Lafayette

Figure 2.13. The *timeline* viewpoint.

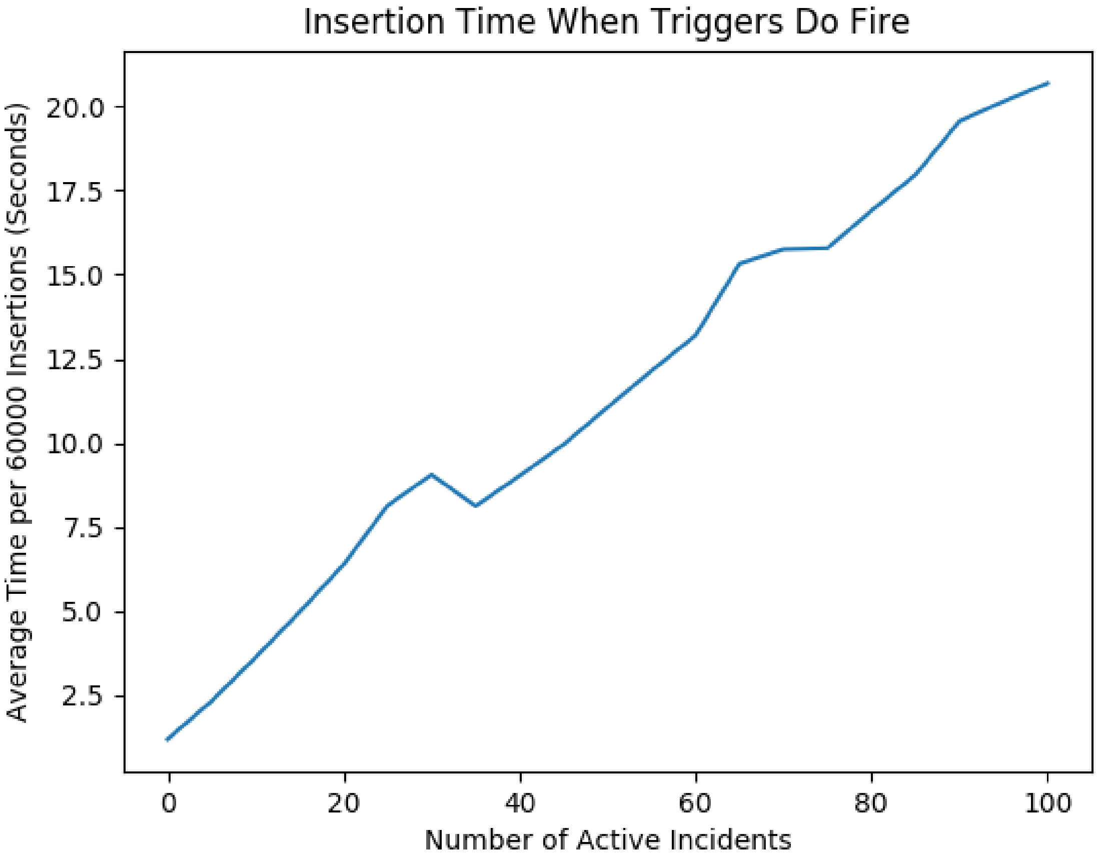


Figure 2.14. Insertion Time vs Number of Trigger Functions. Insertion times are calculated on an average of 5 trials.

Table 2.2. Performance of Different Candidate Sentence Extraction Models based on Clothes Property Identification

	Attr-Only			Attr-Value			θ_H	q_H
Models	Precision	Recall	F1-Score	Precision	Recall	F1-Score		
Word2Vec + POSID	0.83	0.38	0.52	0.85	0.35	0.49	0.5	clothes
RE + POSID	0.86	0.82	0.84	0.92	0.82	0.87	X	wear
WordNet + POSID	**0.93**	0.33	0.49	0.89	0.30	0.45	0.9	*clothes* as noun
SBERT + POSID	0.83	0.49	0.62	0.86	0.45	0.59	0.85	clothes
RE + WordNet + POSID	**0.93**	0.65	0.77	**0.92**	**0.87**	**0.90**	0.9	*clothes* as noun
RE + SBERT + POSID	0.87	**0.87**	**0.87**	**0.92**	**0.87**	**0.90**	0.85	clothes

Table 2.3. Performance of Human Attribute Extraction from Text

Attributes	**Gender**	**Race**	**Height**	**Clothes** Attr-only	**Clothes** Attr-value
Precision	0.94	0.94	0.72	0.87	0.92
Recall	0.73	0.73	0.57	0.87	0.87
F1-Score	0.82	0.82	0.63	0.87	0.90

3. APPLYING MACHINE LEARNING AND DATA FUSION TO THE "MISSING PERSON" PROBLEM

We present a system for integrating multiple sources of data for finding missing persons. This system can assist authorities in finding children during amber alerts, mentally challenged persons who have wandered off, or person-of-interests in an investigation. Authorities search for the person in question by reaching out to acquaintances, checking video feeds, or by looking into the previous histories relevant to the investigation. In the absence of any leads, authorities lean on public help from sources such as tweets or tiplines. A missing person investigation requires information from multiple modalities and heterogeneous data sources to be combined.

Existing cross-modal fusion models use separate information models for each data modality and lack the compatibility to utilize pre-existing object properties in an application domain. A framework for multimodal information retrieval, called Find-Them is developed. It includes extracting features from different modalities and mapping them into a standard schema for context-based data fusion. Find-Them can integrate application domains with previously derived object properties and can deliver data relevant for the mission objective based on the context and needs of the user. Measurements on a novel open-world cross-media dataset show the efficacy of our model. The objective of this work is to assist authorities in finding uses of Find-Them in missing person investigation.

3.1 Introduction

There are many circumstances in which the missing person problem arises. They include amber alerts, family reunification during natural disasters, prison escape, or unaccounted people. Missing person search works similarly for prison escapees, adults with cognitive problems, or missing children. The police have the same problem when they search for a person of interest involved in a crime whether as a suspect or as a victim. For each situation listed above, the authorities have a physical description of the person (e.g., a white male with a medium build, wearing a blue shirt and black jeans) [35]. Physical attributes are used as soft markers for person search [45]. Additional information on missing persons comes from their families, Twitter posts, and phone calls from the public. Available vehicle information can be co-related with Department of Motor Vehicles (DMV) records. Irrespective of the information source, it will have some identifying features of the missing person, based on which the search is conducted. According to related works on missing persons from Policing and Society journal, one of the first steps in dealing with missing person incidents is to search the surveillance camera video footage in the vicinity. For example, West Lafayette, IN, has cameras in all city buses, on many intersections in the downtown area, in the majority of the local business buildings, and in all police cars. Moreover, the policemen themselves are equipped with bodycams when on duty. Police in West Lafayette already spend hours manually searching videos for missing persons [35]. Data fusion from these disparate data sources would be a valuable addition for automatic information retrieval and querying.

In this paper, we report on a system we have built, called Find-Them, to perform video capture, tweets and tips collection, feature identification, and information fusion among these data sources. In Find-Them, we do not attempt to perform facial recognition, as video is low resolution, taken from afar and the lighting conditions are usually poor (due to snow, rain, or darkness). The persons of interest may be in the background or facing away from the camera [45] which makes relying on face recognition infeasible for our task. Instead, we focus on other features, such as gender, clothing (e.g., baseball hat, shirts), and markings (e.g., tattoos). In the absence of the facial recognition, the system is not suited for identification and tracking of a particular citizen, but instead helps with localizing a group of people with similar attributes. While in general the system is not optimized to be used as a digital spy, for the benefit of the society the government should restrict the use of this technology confining it to the law enforcement tasks of searching for the specific persons of interest. Furthermore, objective of Find-them is different from the task of *entity matching*. Entity matching refers to identifying the data instances that refer to the same real-world entity across data sources. Successful systems like Magellan, AutoEM, or CloudMatcher focuses on entity matching by names, whereas Find-them focuses on finding the entities by their physical features.

To begin with, identifying a missing person is a data capture problem. As it was specified before, information about a missing person can come from multiple sources such as surveillance cameras, tweets, family members, and previous occurrences. Storing these multimodal data is a storage problem at scale. Since data comes from multiple modalities and in large amounts, the proposed storage system should normalize different modalities of data at a large scale. Finding relevant information about a specific missing person from multimodal data requires system-specific property identification in each modality and a context-based data integration for a composite query through these modalities. As discussed in [35], training data for property identification is expensive to acquire, and input data from real-world applications often have noise. For the missing person problem, traditional deep learning methods are costly at scale since they require an enormous amount of specific training data. Thus the traditional machine learning methods may fail for extraction of specific features for on-demand missing person identification. Finally, in real-world applications, there are terabytes of information. Therefore, any data fusion has to be done at a large scale while accommodating multiple data sources.

Find-Them implements a streaming data capture and downsampling method to tackle the problem of multimodal data capture and storage. To achieve scalability, Find-Them loads both the raw data and the acquired properties into a Postgres database. Raw data and properties are stored separately between cold storage and an online property server to achieve speed and scalability. We propose modality-specific feature identifiers for video feeds, unstructured text, and tweets. In this work, we explain the feature extractors required for the *missing person* problem. For data fusion, Find-Them implements Entity-Attribute-Relationship schemas compatible with the application domain. Using the features specified by the user, we built SQL queries using the data description language. By performing these queries (e.g., JOIN) over the standard schemas, Find-Them delivers the multimodal results relevant to the user interest. The fusion methodology in Find-Them is expandable to other modalities and different feature identifiers for the discussed modalities.

Firstly, we explain the related works. Then we discuss the Find-Them architecture in detail. In the next section, we describe the proposed feature identifiers with benchmark experimental results. In subsequent sections, we discuss a demonstration scenario for Find-

Them and the generalization capabilities of the system. Finally, we include the future directions and conclusion.

3.2 Related Work

Missing person search is a significant real-world problem that draws on work in several areas of social and computational aspects.

3.2.1 Missing Person Search Applications.

Applications such as People Locator (PL) [46], Myosotis [47], NamUs [1], 'Google Person Finder' allow different levels of missing person search and comparison capabilities. People-locator [46], a search application for family reunification post-disaster, combines multiple modalities for searching and reporting missing people such as structured web form, app-based community reports (ReUnite), unstructured text from email, image-based hospital reports (TriagePic), and other applications with PFIF (People Finder Interchange Format) data format. Similar to Find-Them, for relevance matching PL employed SQL query-based database search and Apache Solr based indexing and search-string matching. However, it lacks the capability of face matching or multimodal searching. NamUs allows to

1. search for matching demographics, descriptors, and distinctive characteristics of a missing person;

2. automatically compare cases based on geography, dates, and physical features and helps to find connections and investigative leads;

3. generate customized case maps.

NamUs has a similar comparison and search functionality as Find-Them. Google Person Finder is a disaster time registry to post and search for missing person status. Myosotis [47] aggregates data from heterogeneous missing people databases, allows visualization via interactive maps, and infers an estimation of the probability of a new occurrence. Neither

[1] ↑https://namus.nij.ojp.gov/

NamUs, nor Google Finder, nor Myosotis allow for multimodal or on-demand search from streaming heterogeneous data.

3.2.2 Person Re-identification (RE-ID).

Person re-id refers to searching for a person in video feeds through a textual or an image query. Existing person re-id methods use both supervised and unsupervised learning [35, 48] techniques. Identity-aware annotations [49, 50] and zero-shot learning have increased the matching performance between image and text descriptions for person re-id by using text attribute query. Attribute recognition in the above models requires a substantial amount of training samples. Multimodal search differs from person re-id in the query-response formats. Cross-modal search allows using different data modalities as a query and as a response as well.

[51] augmented person re-id with facial sketch by fusing the facial attributes and the semantic color information in attributes using a fuzzy rule-based layered classifier. FindThem does not perform any facial recognition, rather re-ids a person via various semantic attributes, including the color information. Existing methods [49, 50] for text attribute extraction considers noun phrases as potential attribute values. [49] filters the candidate phrases using associated images. [50] categorizes the noun phrases to specific attribute phrases such as, *upper-body* following a dictionary clustering approach. These approaches do not consider the noise in streaming documents and the performance bottleneck of parts-of-speech taggers. They also do not differentiate between attribute names and values extraction.

3.2.3 Cross-modal Matching and Correlation Learning.

Most of the previous works [52–55] in multimodal matching have followed the idea of projecting the features from different modalities into a shared embedding space using modality-specific transformations. [52] focuses on correlation learning to learn linear projections using pairwise information. In contrast, [53] uses both pairwise and semantic information, e.g., class labels, to learn the common subspace. [55] extends deep canonical correlation analysis with an auto-encoder regularization term for nonlinear representations of multimodal data objects. Peng et al. [54] better encodes the intra-modality and inter-modality correlation with hierarchical networks.

Some recent methods learn richer semantic representations for different modalities by using attention mechanism [56], graph representations [57], and generative models [58] to build the encoding networks. Deep Relational Similarity Learning [59] avoids explicitly learning a common space by integrating relation learning, capturing the implicit nonlinear distance metric.

While these learning methods exhibit good performance, mainly on bi-modal datasets, they require a large amount of training data and do not scale well. Data representations lack generalization capability across multiple modalities or data sources. Besides, many existing application domains already have pre-derived domain-specific features with fine-tuned feature learning methods, but the above models cannot integrate these sources. Moreover, current metric learning methods can only integrate user-specified data relevancy with training samples or with class labels. The data fusion methodology described in Find-Them focuses on solutions for the problems of scalability, lack of annotations, and use of pre-identified features for data fusion.

Data fusion among multiple modalities has been used in many application domains such as sentiment analysis [6], image-text matching [57], face retrieval [51], and visual question-answering for a better understanding of context. These approaches have performed well for respective application domains, but they lack generalization capabilities. Similar to Find-Them, [60] built a multimodal relational knowledge base by continuously querying for detected objects from videos and matching objects in text. However, they do not perform any attribute-specific search and cannot be generalized for multimodal person search.

3.3 System Overview

Figure 2.1 illustrates the architecture of Find-Them. Find-Them is divided into four modules *data ingestion*, *feature identification*, *relevance modeling*, and *data retrieval. Data ingestion* deals with the problem of data capture and data storage. The system captures the streaming data and loads it into Postgres at the server end after necessary pre-processing.

Feature extraction is done during load time using type-appropriate models for each data source. Extracted properties are inserted into Postgres following the schema determined by the entity-attribute-relationship model. The defined schema is used to create data integration among multiple sources during the *relevance modeling* phase. Users issue one-shot and standing queries to the system in the *data retrieval* phase. *Ingestion* and *retrieval* systems can operate in parallel. A user preference model is built from the history of user queries and is used in conjunction with the relevance model for data retrieval.

3.3.1 Data Ingestion

3.3.1.a Data Capture.

In Find-Them, we employ a streaming data capture system for video, unstructured text, and tweets. While capturing tweets, we filtered the tweets with hashtags (#wetip, #FultonMissing) and user profiles (@CambMA, @WLPD). We utilize Twitter search API to find tweets with a specific hashtag or user-id from historical tweets. Streaming API captured the streaming tweets matching the search tag. Finally, we deploy Kafka to ingest them into the Postgres database to keep missing person cases separated by using each case as a topic, as seen on the *data capture* module. Kafka consumers read from the topics and store the JSON output from the API to Postgres. The tweet pre-processing module also uses the JSON output as input. Using Kafka to read from each case separately ensures parallel processing of multiple missing person cases.

For each modality, we adapt a different pre-processing system with a high-level property identification. The extracted properties are chosen based on the requirements of the application domain. This additional feature identification step is done at load time to reduce response time during a complex query. Subsequent feature identification stages use the output from the pre-processing steps as inputs. The granular features are more complex and often involve computational overhead. Hence, we extract these features on-demand. For example, for missing persons, authorities are looking for human attributes, so *people* are identified during data ingestion for video feeds. In later stages of the feature identification, we extract different properties of a person, such as, gender, race, cloth colors.

3.3.1.b Pre-processing of Video Feeds.

Find-Them follows similar ingress steps as SurvQ [35] for video feeds. When the videos arrive at the server in real-time or as a bulk manual upload, they are converted to MP-4 from their current format and are downsampled to one frame per second for further processing. YOLO [2] is applied to each of these frames to identify the *objects* described in the Pascal VOC dataset (http://host.robots.ox.ac.uk/pascal/VOC/). For high-level object detection, Find-Them uses YOLO because of its run-time efficiency and availability of pre-trained models with a large number of object classes. The Pascal VOC dataset includes 20 class labels, including *person* and seven types of vehicles, making it a good candidate for the pre-trained model in the missing person problem. Each YOLO-detected object is further examined in the feature extraction stage to identify finer granularity *object properties*.

3.3.1.c Pre-processing for Unstructured Text and Tweets.

Documents are converted to *plain text* from their incoming formats. Pre-processing module standardizes texts in the documents by removing jargon, articles, abbreviations, and short forms of regular English words, depending on the source of data collection. The remaining texts are converted to lower cases.

The result from the Twitter API comes with a lot of metadata, which is helpful during data fusion. Raw JSON object outputs from the API are parsed to separate the metadata and original text. Texts in tweets are similar to unstructured text but include jargon, hashtag, user tags, and abbreviations. So, before processing the tweets as documents, text in the tweets is cleaned after removing or replacing the jargon with the closest English words. As the next step, hashtags and user tags are removed from the text. Feature extraction module designed for documents takes these cleaned and parsed texts as inputs.

Find-Them has an extendable library of feature extractors for video and text. We explain the extractors needed for the *missing person* problem in detail in the *Feature Extraction* section, along with the experimental results used for validation on datasets from real-world applications. However, Find-Them is extendable to other modalities and feature extractors. Feature extractors for other modalities can be added and used in a plug and play mode in

Find-Them. It is also possible to use different feature extractors than the ones used in this paper, given that they have the same output features.

3.3.1.d Data Storage.

To achieve scalability and a faster response time, we store the outputs of the feature extractors in separate Postgres tables for each modality, along with pointers to the archived raw videos and texts. Tweet metadata and user metadata are stored in different tables. This solution allows finding relevant data objects with SQL queries in real-time.

3.3.2 Relevance Modeling and Data Fusion (EARS)

3.3.2.a Entity-Attribute-Relationship Model with Schema Mapping.

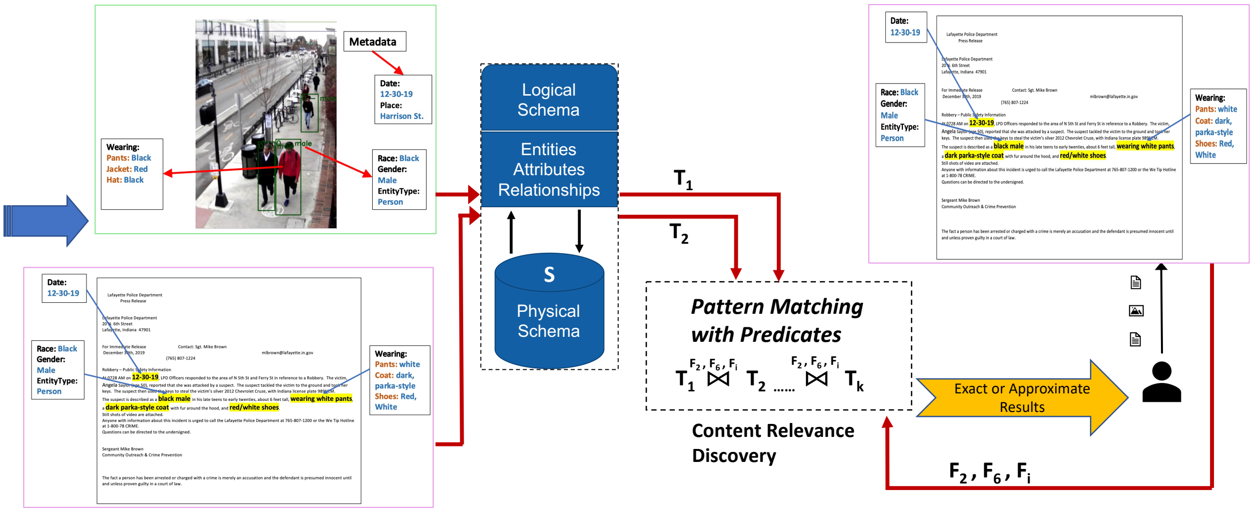

Metadata
Date: 12-30-19
Place: Harrison St.
Wearing: Pants: Black Jacket: Red Hat: Black
Race: Black Gender: Male EntityType: Person
Date: 12-30-19
Race: Black Gender: Male EntityType: Person
Wearing: Pants: white Coat: dark, parka-style Shoes: Red, White
Logical Schema
Entities Attributes Relationships
S
Physical Schema
T_1
T_2
Pattern Matching with Predicates
$T_1 \overset{F_2, F_6, F_i}{\bowtie} T_2 \ldots\ldots \overset{F_2, F_6, F_i}{\bowtie} T_k$
Content Relevance Discovery
Exact or Approximate Results
F_2, F_6, F_i
Date: 12-30-19
Race: Black Gender: Male EntityType: Person
Wearing: Pants: white Coat: dark, parka-style Shoes: Red, White

Figure 3.1. Data Fusion for Relevant Data Recommendation

For real-time data fusion, we propose to construct an entity-attribute-relationship (EAR) model for each application domain and then map to a relational database with schema S, as shown in Figure 3.1. Each source needs to follow this defined schema. Adding a new data source to the system would require extending the EAR model and the schema.

For example, Figure 3.2a and Figure 3.2b show the individual schema of incident reports and videos for the problem of person identification for the West Lafayette Police Department.

In Figure 3.2c, we show the proposed combined schema for cross-modal retrieval for mining relevant data objects describing the person of interest. We translated all extracted features from video and text to the above schema during data storage.

3.3.2.b Data Fusion with SQL JOIN.

We propose to use the **E**ntity-**A**ttribute-**R**elationship model with **S**QL querying (EARS) for data fusion. Since data from each source has the same schema after schema mapping, matching between data objects of different modalities translates into JOIN queries between the tables. The results can be presented as an exact or approximate match depending on the conditions imposed on the JOIN query.

We implement a nested loop join on relations from each modality and the incident relation. Each queried missing person incident are converted into relation R with features $F_1, F_2, \ldots, F_m$. Features from modalities are translated into relations $T_1, T_2, \ldots T_n$, where n is the number of modalities in the system. We perform a join between R and each T_k ($k \leq n$) using the join predicate JP on all queried features,

$$JP(T_k, R) := \sum_{1 \leq \mathrm{i} \leq m} (T_k.F_\mathrm{i} == R.F_\mathrm{i}) \tag{3.1}$$

For example, in Figure 3.1, features from the video feed are translated into relation T_1, and features extracted from the incident report are translated into relation T_2 after schema mapping. If the user is interested in a person with features $F_2, F_6, \ldots, F_\mathrm{i}$, we create a JOIN query over all the translated relations and the incident relation on features $F_2, F_6, \ldots, F_\mathrm{i}$.

3.3.2.c User Preference Modeling.

Find-Them employs a simplified user preference modeling to keep track of changes in user requirements. We keep a record of the historical queries made by the user. For now, we issue notifications during the streaming data delivery only for the current user query. For future improvements, we are building a predictive model using the history of the user queries. This model will ensure a better on-demand data delivery and creation of notifications based on both the context and the current user's query.

3.3.3 Data Retrieval

During data retrieval, Find-Them expects a user to either create a missing person incident or upload an example video/image/document/flyer (Figure 3.3) that describes the missing person. As seen in Figure 2.10, for incident creation, the user will upload gender, race, upper body color, lower body color, and head/hair color as a description of the missing person. Users will also mention the date range and the search area they are interested in searching through.

In the former case, the example is parsed using the modality-specific feature extractor, and the extracted features are used as user inputs. As seen in Figure 2.1, features mentioned by the user are considered as predicates to SQL queries and are defined as triggers to the Postgres DBMS. Using one-shot and standing queries allows us to find the desired result from both historical and streaming data. One-shot queries are immediately translated into SQL for schema S and are executed. Standing queries are handled by triggers, which are invoked automatically when any matching data arrives. When the queries involve information from one modality, the retrieval is straightforward. If similar data arrives in the future from other modalities, the trigger associated with the fusion model will link them and deliver the streaming data objects as standing query results.

3.4 Feature Extraction

Our primary use case for the missing person was person identification for West Lafayette Police Department (WLPD). WLPD searches for missing persons and suspects in a similar way.

Persons of interest are described with different physical attributes, such as gender, race, physical build, height, hair color, color and description of their clothes, and other visible features in their body. These descriptions are circulated through press releases and missing person flyers. Whenever there is a 9-1-1 call, the authorities generate an incident report describing the series of events. After the due investigation, the involved officers write an investigation report on the incident. Both of these reports include person descriptions, as mentioned above. We analyzed the text in the incident and investigation reports shared by WLPD with us. WLPD shared these reports after proper anonymization of identifying information. The top frequencies of different attributes for person profiling in the documents are as follows: almost all documents use gender and race, 78% of the reports use clothes (such as shirts, jeans, pants, jackets), and around 57% use height to describe a person. Therefore, in this work, we only describe the feature identifiers that were used to extract gender, race, clothes color in videos and text, as follows:

- For identifying clothes colors and tracking a person in the video feeds, we used a heuristic-based *color sampling* method [35] with YOLO as the background object identifier. This method allows us to identify and track a person only based on external identifiers without violating privacy.

- For gender and clothes detection in videos, we relied on the traditional deep learning object detection method and re-trained YOLO with newer class labels.

- For gender, race, and clothes details detection in unstructured text, we used the *HART* model based on regular expression search, Word2Vec embedding, and pattern recognition. We also used a topic-based similarity search technique for finding tweets or texts describing the objects in the videos. Using text embedding allows us to identify the ambiguities in different people's writing style when describing color.

The feature identification in visual modalities is significantly different to the feature identification in textual modalities. Since text modalities describe the color of clothes in words, there can be ambiguities in the description. On the other hand, in videos, the colors can have high variance ranging from light to dark. The extracted features are stored in Postgres following the common EAR model, which allows us to perform uniform SQL queries across different modalities. We have benchmarked these models on real-world datasets and used the extractor results during the data fusion.

3.4.1 Color Analysis for Body Details.

For color sampling [35], we use the bounding box of *persons* from the YOLO detection. The bounding box is segmented into three body parts - head, upper section, and lower section. We segment the body parts by estimating the ratio of each part to the bounding box according to the human body proportions in anatomy. First, RGB values are extracted from each pixel in a segmented region. Colors for each segment are assigned by calculating the smallest distance between the extracted RGB values and the standard RGB values. Integer RGB values make it easier to compare the extracted colors to baseline colors. In the case of multiple colors in a region, a majority voting is applied to determine the color of the area.

WLPD-Video-Dataset. We have collected and labeled over 20 hours of video from different cameras and locations in the city of West Lafayette. Six custom classes with over 12200 images were labeled manually for re-training and testing the YOLO network to detect gender, clothes, and color. Each one-minute chunk of the video consists of around 20 frames, sampled at 3-second intervals.

In the test set from WLPD-Video-Dataset, the clothing colors were recognized with high precision, while the color of the sampled head area were more prone to be affected by the color of the background, as shown in Figure 3.4.

Based on the color information, we can trace the movements of pedestrians across continuous frames. Figure 3.5 shows the moving routes of two pedestrians walking towards each other. The dotted line after each pedestrian indicates their moving direction.

In cities, multiple cameras are installed at the same traffic cross-section to observe pedestrians from different angles, with each view providing additional information. We wanted to trace the same person across multiple cameras installed at various locations for the missing person search. Figure 3.6 shows two examples of tracking the same pedestrian passing through three areas. On the left, we track a cycling person wearing a red shirt passing from locations 1 to 3. It takes only 39 seconds since he is cycling. On the right, we follow a walking person wearing a red shirt passing from location 3 to 1 in the opposite direction. It takes him about 6 minutes. So we can map out the walking trajectory of a person as long as there is no change of clothes.

3.4.2 Re-training YOLO.

For gender and clothes detection in video feeds, we re-trained YOLO [2] to identify gender and clothes in video feeds. Hue, saturation and brightness (HSB) for each frame has been analyzed to improve object detection and recognition under night and changing weather conditions. The range of the HSB values are tracked for each color as time passes and the updated values are used for more accurate object detection and recognition. We are building fine-tuned YOLO models for future improvement. We report results for both gender and cloth detection with YOLOv3 and YOLOv4 in Table 3.1. For gender and clothes detection, we achieved 68% mAP and 67% mAP, respectively, when YOLO is re-trained without pre-trained features. Achieving higher performance with real life low resolution raw video under different light and weather conditions is a difficult task which requires future work.

Table 3.1. Performance of YOLO for Gender and Clothes Detection in WLPD-Video-Dataset (mAP)

Object	**YOLO v3**	**YOLO v4**
Gender	0.59	0.68
Clothes	0.56	0.67

3.4.3 Human Attributes from Unstructured Text.

Using the stacked (Regular Expression (RE) + Word2Vec) variant of the HART model [61], we identified Candidate Sentences (C_s) from the texts of cleaned documents and tweets. We searched for *clothes* with regular expressions on the sentences for finding C_s. If it returns no result, the problem is formulated as a similarity search among all tokens in a sentence, where *clothes* are used as the search token. We used the pre-trained Word2Vec embedding for each token as features. If the cosine similarity between any token in a sentence and the search phrase reaches an empirical threshold, we consider it as C_s. For the attribute value detection from C_s, specific patterns were searched for recognizing gender and race. For clothes identification, we followed the *Clothes Name and Value Identification* algorithm from [61] which uses Parts-of-Speech (POS) tags of tokens to identify the description.

FemmIR-text Dataset. For benchmark results on text features, we used part of the text data from [61] consisting of incident reports, press releases, and officer narratives from historical cases. It contains 13 press releases, 40 officer narratives, and five incident reports. Due to privacy reasons, WLPD publicly released only a subset of redacted reports.

Table 3.2. Evaluation of Human Attribute Extraction on FemmIR-text

Attributes	**Gender**	**Race**	**Clothes** Attr-only	**Clothes** Attr-value
Precision	0.94	0.94	0.93	0.92
Recall	0.73	0.73	0.65	0.87
F1-Score	0.82	0.82	0.77	0.90

For unstructured text, as seen in Table 3.2, the HART model performs adequately for an on-demand detection model. Results for clothes are reported for (RE + Word2Vec + POS) model on two evaluation metrics, attribute-only and attribute-value.

3.4.4 Semantic Similarity Search by Topic.

We employed a topic-based similarity search to extract documents describing the same objects and attributes found in videos. We also used it as an additional method for finding candidate sentences. Assuming that each sentence in a document is a mixture of some topics, if any of those topics explain the search phrases, we posit that the sentence is a Candidate Sentence. We used Latent Dirichlet Allocation (LDA) to identify the hidden topics of the sentences in the documents and the query phrases (e.g., clothes, car, person, male). LDA is a generative topic modeling technique where documents are represented as random mixtures over unseen topics, and the topics are derived by calculating distributions over all the words in the document. In this case, we have represented each sentence in the document and the query phrase as individual mixture of topics. For distribution measurement, term frequencyinverse document frequency (Tf-idf) vectors of all tokens in each sentence were used as unigram features. The cosine similarity of the query phrase topic against the topics of the corpus of sentences measures the closest sentence matching the query. We have collected 249857 tweets from 77943 users describing topics related to Cambridge, MA in the *Cambridge-Public-Authority-Tweets (CPAT)* dataset.

3.5 Demonstration

Finally, we demonstrate Find-Them on the incident reports, press releases, and video feeds from the West Lafayette Police Department. We are working on adding the Department of Motor Vehicles records and public tips as additional data sources in the future. We show how Find-Them can accurately detect and track a missing person based on non-invasive physical properties and minimize the investigation effort to find a missing person. We describe the users interaction through six steps, impersonating an officer in WLPD. We annotate each step with a circle in Figure 3.8.

Step ① (Create Missing Incident or Upload Example): First, the user uploads an incident report, a flyer, or a tweet with a physical description of the missing person with the search area and the search timeline in step (1b). They can also upload a video clip or snapshot of the missing person. In this case, we apply appropriate feature extractors to the examples based on their modality. Then the predicates for the search query are created with the extracted features. When the user does not have any examples, they can create a missing person incident by filling out the person's details, the search area, and the timeline, as seen in step (1a).

Step ② (Creation of Predicates): For searching a person, the WLPD officer specified the identifying properties in step 1. Using those inputs, we created an *incident* schema which becomes the search criteria for current and future streaming data in step (2). Triggers in Postgres await for streaming data with similar features to the *incident*, and it notifies the police officer of any matching video feed or tweets. The police officer can always revisit the *incidents* from their search history.

Step ③ (EAR Mapping). As seen in Figure 3.2a, incident reports have a feature extractor that outputs clothes as individual entities and then extracts their details, whereas,

in Figure 3.2b, we can see the details are extracted in terms of body parts. Both of these modalities need to map to the common EAR model shown in Figure 3.2c. The system maps the incoming document features to the common EAR model as follows -
(shirts, jackets) → upper body,
(pants, jeans) → lower body, and
(hat, cap) → head.

Step ④, ⑤ (JOIN among Data Sources). Before this step, data from each modality is stored in Postgres tables in an atomic manner. The data storage schema was built considering different categories of features necessary for missing person problems. For example, physical details about a missing person were saved in one table, whereas incident location was stored in another. Separation of storage allows us to answer simple queries requiring only one type of information quickly. When the query involves multiple types of information, we create SQL queries to perform JOIN between separate tables representing features from different modalities. The first JOIN in step (4a) separately creates the primary results from each modality. In step (4b), we performed a union of all modalities. Finally, in step (5), we perform a JOIN between the accumulated results and the previously created *incident* table to extract the subset of data objects which match the user search criteria and show the multimodal result on the investigation page.

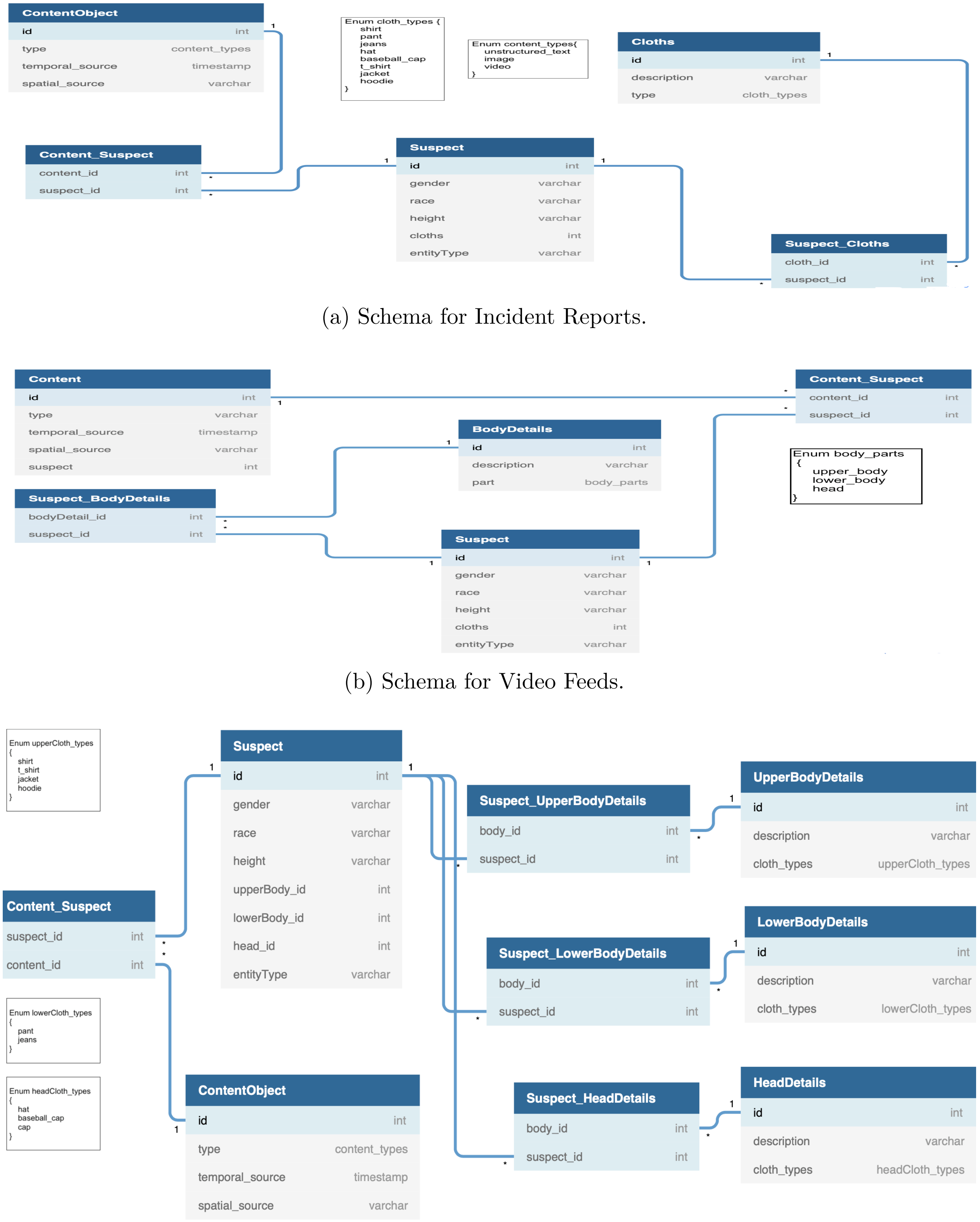
Step ⑥ (Different Viewpoints). Similar to [35], there are three possible viewpoints the user can choose from to see the results- list, map, and timeline. The timeline view was generated to mimic the investigation timeline, whereas the map view allows us to pinpoint a location. The user can also choose their favorite results and can see the filtered result at a later time.

3.6 Scalability, Universality, and Multi-user.

Find-Them establishes a common information model, *relational schema* across multiple data sources, and eliminates the need for separate data representation and linking methods. These models are universal for all modalities without additional overhead since converting features into relational tables is a linear process. The linking process for EARS can scale to a large number of properties from data objects, and EARS does not require any training. The system demonstration shows that we could query historical data (in thousands of records) and streaming data in real-time during inference time. For the space constraint, we do not include the time comparisons here. Find-Them is capable of extension to multiple users, each with their own set of preferences in the form of queries and data objects. Since each user has a mapping to the retrieval set with their queries, their queries are kept separate.

3.7 Conclusion and Future Work

This paper has introduced Find-Them, a feature-based multimodal data fusion system for analyzing video feeds with other data modalities for finding missing persons. We have described a database backend, along with a schema and a relational query-based fusion method that can scale to a considerably large amount of data volume, along with a fast response time. Our experimental results showed satisfactory performance for the feature identifiers for commonly used missing person features. Find-Them can also identify the connections between historical and incoming missing cases, giving the law enforcement officers an edge in their investigations. In the future, we will expand the video and text datasets by including mobile camera videos, city maintenance records, and Bureau of Motor Vehicles records. We also have future goals to include more data modalities and evaluate the effects of humans-in-the-loop on improving performance. We further benchmark the EARS algorithm for searching a person with certain features in an incremental work. In future work we will test Find-Them and its viewpoints capability at rush hours with data collected during game day with heavy traffic in the university area and manually annotated map-timeline ground truth. Finally, we will extend the framework to include feature extraction as part of the relevance modeling in an end-to-end neural network architecture and user interests will be modeled based on their historical queries.

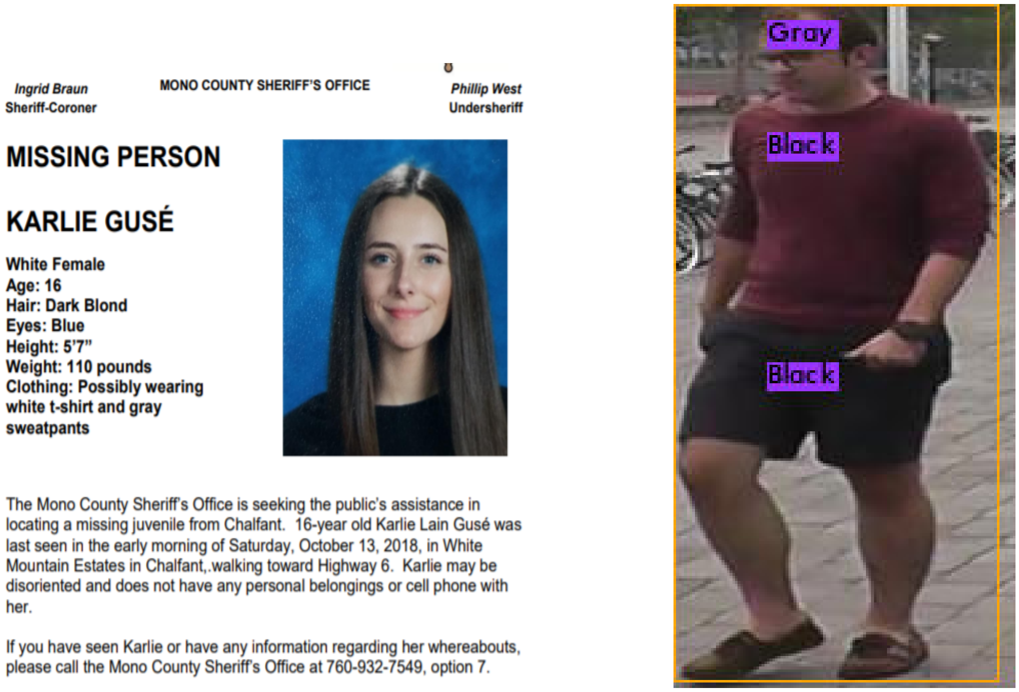


(a) Schema for Incident Reports.

(b) Schema for Video Feeds.

(c) Combined Schema for Fusion between Multiple Modalities.

Figure 3.2. Schema Models for Data Storage.



Ingrid Braun
Sheriff-Coroner

MONO COUNTY SHERIFF'S OFFICE

Phillip West
Undersheriff

MISSING PERSON

KARLIE GUSÉ

White Female
Age: 16
Hair: Dark Blond
Eyes: Blue
Height: 5'7"
Weight: 110 pounds
Clothing: Possibly wearing white t-shirt and gray sweatpants

The Mono County Sheriff's Office is seeking the public's assistance in locating a missing juvenile from Chalfant. 16-year old Karlie Lain Gusé was last seen in the early morning of Saturday, October 13, 2018, in White Mountain Estates in Chalfant,.walking toward Highway 6. Karlie may be disoriented and does not have any personal belongings or cell phone with her.

If you have seen Karlie or have any information regarding her whereabouts, please call the Mono County Sheriff's Office at 760-932-7549, option 7.

(a) Flyers. (b) Screenshots.

Figure 3.3. Upload Example of Data Objects.

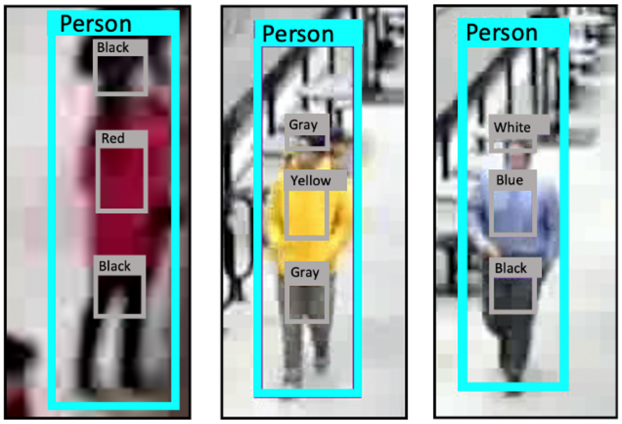


Figure 3.4. Color recognition on *WLPD-Video-Dataset*

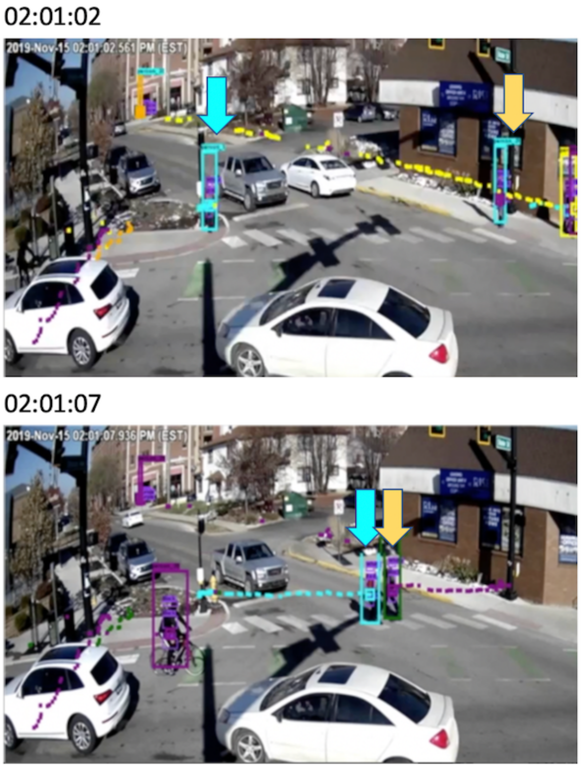

02:01:02
2019-Nov-15 02:01:02.561 PM (EST)
02:01:07
2019-Nov-15 02:01:07.936 PM (EST)

Figure 3.5. Tracking of a pedestrian crossing the street from a singular camera.

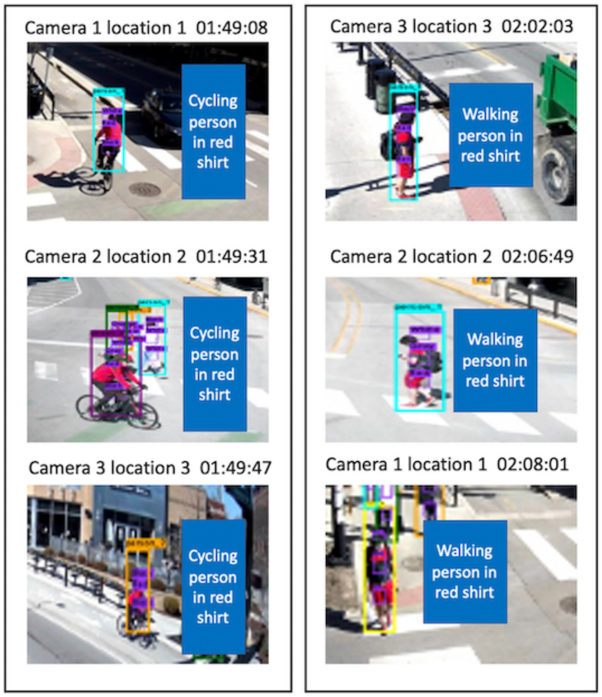

Camera 1 location 1 01:49:08
Cycling person in red shirt
Camera 2 location 2 01:49:31
Cycling person in red shirt
Camera 3 location 3 01:49:47
Cycling person in red shirt
Camera 3 location 3 02:02:03
Walking person in red shirt
Camera 2 location 2 02:06:49
Walking person in red shirt
Camera 1 location 1 02:08:01
Walking person in red shirt

Figure 3.6. Pedestrian tracking at multiple scenes with multiple cameras.

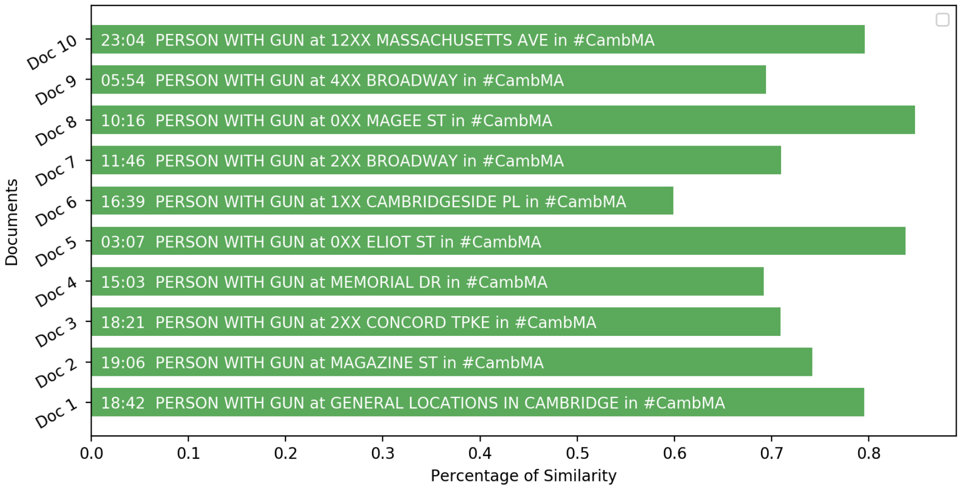
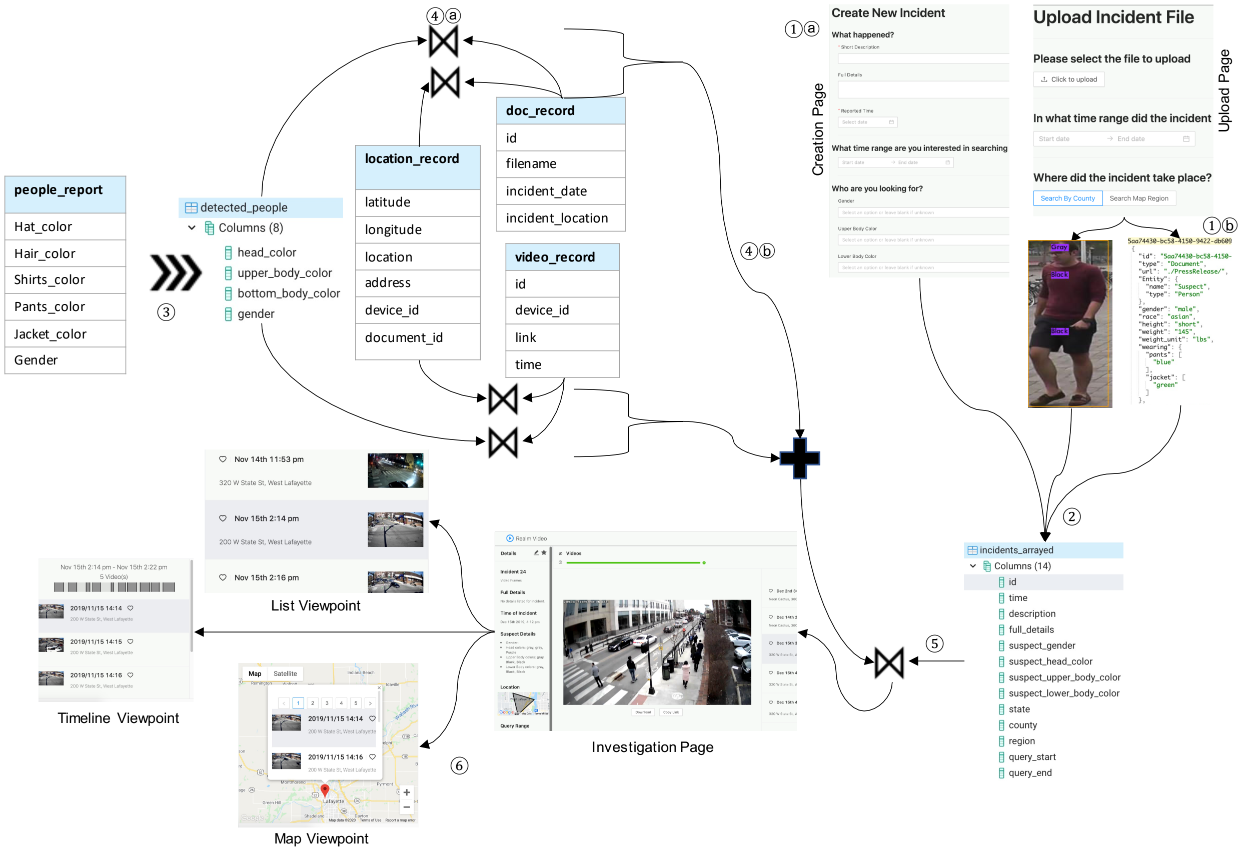


Figure 3.7. Relevant Tweets with LDA in the CPAT dataset describing a *PERSON WITH GUN* in *Cambridge Area*.



```
5aa74430-bc58-4150-9422-db609
{
  "id": "5aa74430-bc58-4150-
  "type": "Document",
  "url": "./PressRelease/",
  "Entity": {
    "name": "Suspect",
    "type": "Person"
  },
  "gender": "male",
  "race": "asian",
  "height": "short",
  "weight": "145",
  "weight_unit": "lbs",
  "wearing": {
    "pants": [
      "blue"
    ],
    "jacket": [
      "green"
    ]
  },
```

Figure 3.8. Find-Them Demo.

4. FEATURE-CENTRIC MULTIMODAL INFORMATION RETRIEVAL (FemmIR)

4.1 Introduction

With the influx of media collections, exploratory data analysis requires comparing data from different modalities to grasp a more informed decision for any phenomenon. With the ever-growing size of the multi-media data, multi-modal data analysis becomes difficult for any real-world application-specific information needs, specially when there is heterogeneity among data properties in different modalities and information needs. Current data discovery systems rely on manual lookup, exploration of the relational database structure, or cross-modal information retrieval for the data preparation task. Traditional cross-modal retrieval models create a common representation space to compare the similarities between data sources, whereas relational query models find the user intentions from the query history and deliver the data tuples that match the user preference model. However, having a common subspace to translate all incoming data, or designing new queries for every new modality causes a system with existing properties to fail. So we ask the question, *how can we handle data retrieval in applications with existing object properties and explicit information needs?*

Two common reasons retrieval systems with application-specific information needs fail are: (1) *disconnect* between high-level information needs and low-level object properties, and (2) lack of annotated data compared to the size of the multimedia data. Object properties are often specified to the system as high-level information needs, whereas most retrieval systems use representation models to gather low-level features from different modalities before mapping them into the common subspace to compare them. Multi-modal systems that can process high-level information need described as properties [1, 62] often have to handle retrieval from a large repository, or streaming data. [1] expects to process 60,000 frames per minute from the camera feeds [35], whereas on average 6,000 tweets are generated per second in [1]. Most retrieval systems cannot process data ingestion for these large amounts of data, and annotating them for training to discriminate between relevant and irrelevant, is a near-impossible task.

Therefore, in case of application-specific information needs, one should explore a retrieval system that would use existing object properties in the data. Examples of commonly observed retrieval system failures caused by explicit information need include: (1) decline of model accuracy (due to using the same property identifiers for all systems, and lack of generalization), (2) waste of computational resources (due to re-running identifiers), (3) excessive processing time (due to failure to scale to large data), (4) inability to incorporate new modalities (due to incoherent representation and redundant properties), and (5) system crash (due to invalid modalities beyond design). These examples indicate a common problem: *mismatch* between properties in the information need and in the data, as well as *dependence on annotations*. To motivate our work, we start with an example inspired by a real-world system, where local law enforcement officers asked for assistance to sift through hours of videos [35].

Object-property focused Information Need *An agency wants to build an automated system to find persons of interest from many hours of video feeds. Incident reports and text queries were considered to be text modalities. Alex is asked to develop a machine learning (ML) pipeline over this dataset to predict the videos where the person mentioned in the text would be found, and, subsequently, the authority would look for them in those videos. Alex decides to use an off-the-shelf retrieval algorithm that is trained over video and text multi-media data. But the performance was not satisfactory. Alex was not able to modify the model to focus on specific properties which are most common for a missing person. On the other hand, he could not run transfer learning as the annotated data is very difficult to achieve in this case, where one positive case occurs in 8-10 hours of video. Now he wonders: (1) how can he re-train the retrieval model without any training data to focus similarity on the desired features? (2) If he runs a property identifier in each data modality and performs only explicit matching would that achieve the desired performance? (3) How can he map similar properties from each modality?*

Existing tools [63, 64] that use encoder-decoder architecture or metric learning to map low-level features to a common space cannot explicitly consider high-level properties. Also, the system required a soft-match approach rather than an exact match since finding persons-of-interest is a sensitive use case, and although organizations want to ease their workload, they do not want to commit any mistakes. Example 4.1 is one among many incidents in real-world applications where similarities among multi-media sources are required with a focus on specific object properties [citations]. As mentioned in prior work [65], *“21% of the bugs encountered in Microsoft Azure services were due to inconsistent assumptions about data format [66]. Furthermore, 83% of the data-format bugs were due to inconsistencies between data producers and consumers, while 17% were due to mismatches between different consumer interpretations of the same data. Similar incidents happened due to misspelling and incorrect date-time format [67], and issues pertaining to data fusion where schema assumptions break for a new data source [68, 69].”* Hence a retrieval system must handle the inconsistent assumptions about object properties from different data sources. We provide another example where a system fails with the introduction of novel modalities or data sources.

Mismatched Properties across Data Sources *As pointed out as an example in prior work [70], “An organization within the Air Force collects data from sensors to support data scientists in producing data-driven reports for decision-makers. This vast and heterogeneous data is organized across hundreds of tables in a data lake, each with a different schema.”* DICE [70] helps in finding the right data sources by finding join paths across tables and involving human-in-the-loop. But as new tables from different sensors are added, there is no guarantee the previous joins will hold.

The aforementioned examples bring forth three key challenges. First, we need to find a *common representation model* for object properties from all modalities and *map* them into a common embedding space for the downstream similarity matching task. Second, the similarity of the data samples needs to be *measured* in a manner that captures the *approximate matches.* For example, two records can be similar if they have the “same entities”, and/or they describe the “same event”. Third, for the retrieval model, we need to find a *training* method in absence of annotated data.

Common representation modeling with Graphs. Towards solving the first challenge, our observation is that most real-world data describes relational knowledge among different entities, along with their attributes and metadata such as spatial and temporal data. *Graph representation* allows these structural and characteristic information to be stored and accessed efficiently [71], across multiple modalities. Besides, deep learning-based dynamic graph embedding methods [72, 73] learn low dimensional vector representations for the graph while preserving both the graph properties and the structure. Despite different naming and organization conventions across various data sources, each data-sample still holds the relationship properties among entities. This eliminates the issue of heterogeneous property representations. For example, *social network recommendation engines* (such as, Yelp [74], or Pinterest [75]), or *multi-media recommendation applications* (such as, micro-video recommendation in tiktok, Kwai, or MovieLens [76]) often use graph representations.

Tensor-based similarity comparison. Our second observation is that real-world information need often emphasizes implicit matching (retrieval matching, entity-relation matching, or user-item matching) [77] rather than just explicit matching. Since tensor is a geometric object that describes relations between vectors and prior works [78] have shown that neural tensor network (NTN) can explicitly model multiple interactions of relational data, we choose to use an NTN-based framework to measure the similarity between graph representations. The optimal number of interaction scores is application-specific. For example, *an application looking for person-of-interests wants to soft-match a person with similar race, gender, and clothes, whereas a missing person search would require to match all human attributes for an exact match.* The model would learn that the multi-media samples from different modalities describing the same entity would be in similar parts of the semantic space.

Weakly supervised learning for retrieval. Our third and final observation is, capturing how much change is needed to convert one data-sample to another can provide us with a source of weak supervision for cross-modal retrieval. We consider a data sample as a collection of objects with certain properties along with the relationships among them. Since graph edit distance (GED) has been shown to be an effective graph distance metric in many applications, such as graph similarity search [79, 80], graph classification [81, 82], image indexing

[83], etc., we modeled a data-sample similarity metric based on GED. Since multi-media data is usually large in number, it is expensive to annotate the relevance for each different system. Prior works in retrieval system [84–87] use inexact weak supervision to tackle this issue of annotation expense.

Solution Sketch. We propose FemmIR, a framework that compares data from different modalities and heterogeneous sources to user-provided information need and calculates a similarity score among them. Our framework involves three main components:

1. *Data Ingestion:* a graph encoding mechanism that translates properties from incoming data into an attributed graph representation. In general, property names are considered as edge labels, whereas values are used as node labels.

2. *Weak Label Generation:* For capturing the similarity between a pair of data samples, we define a new distance metric that indirectly holds the entity and relationship constraints between the samples, Content Edit Distance (*CED*). CED captures the amount of change needed to convert one attributed data graph to another by including the object replacement cost for cost matrix calculation in Munkres algorithm [88]. CED is later used to define **relevance** label based on system requirements.

3. *Similarity Comparison:* Finally, we train a lean-able embedding function for multiplicative comparison between attributed graphs using the SimGNN architecture [89]. During the training, the objective function minimizes the difference between the predicted score and the ground truth obtained from converting the *CED* into a similarity score. During inference, the learned embedding function calculates the similarity score between the attributed graphs from the data-records.

Given a scenario where the user provides information need as an example and incoming streams have identified properties, FemmIR starts with building the attributed graphs. In case of unseen raw data, FemmIR extracts the object properties either offline, or with priority polling [35] for bulk streams in an additional *property identification* component. Second, the CEDs between the graphs are calculated between the query example and the data samples. Finally, the model is trained to calculate the similarity score between records.

Scope of our work. In this work, we only focus on retrieval cases where either (1) properties from different objects and relationships among them have been identified, or (2) the system has specified its own identifiers for specific data modalities. Note that prior data-matching approaches [1, 90] that employ retrieval model on high-level properties assume there exists a common schema or feature mapping among different modalities. In contrast, FemmIR is agnostic to the design of the source-schema and can support any type of property schema from any data source ranging from raw data in data warehouse (Example 4.1) to a relational database in a data lake (Example 4.1). FemmIR also delivers varying degrees of relevance without the computational overhead. However, FemmIR cannot handle multiple query examples at the same time for single information need as it requires predicting user intent from those examples [65] and that is not the focus of this work.

FemmIR requires knowledge of the application-specific property identifiers to be used throughout the system. The choice of property identifiers depends on the domain knowledge and the properties-of-importance, e.g., *in Example 4.1, Alex would require identifiers for video and text which extracts human properties i.e., gender, race, cloths-worn, cloths-colors, etc.* This assumption holds because: (1) for object and action recognition tasks there exists a well-known set of relevant identifiers for common modalities - video [2, 91], text [92], image [93, 94], and 3D models [95] with reasonable performance. Objects and actions cover the most common properties in retrieval applications. (2) If the information need is expressed through high-level properties [1, 96, 97], we can assume the system already extracted properties from most modalities and domain experts are typically aware of the likely class of properties for the specific task at hand and can easily provide this additional knowledge to the system.

Property Identifiers. While we use the property-identifier outputs to find relevant data sources to the query example, developing identifiers is orthogonal to our work. A number of object and action recognition paradigms from different modalities exist in the literature. FemmIR assumes access to a suite of property-identification techniques and uses them to extract properties from the data. To support a new data source, FemmIR needs to know the corresponding modality and the properties-of-interest. We discuss some common classes of property identifiers as representative ones, which are currently supported in the implementation of FemmIR. We have selected them based on the criteria of having semantically similar properties or similar property definitions. For properties discovery in visual modalities, we rely on prior work on action recognition [91, 93], object detection [94, 98], etc. As part of FemmIR, we proposed a novel property identification method for properties described in textual modalities. Properties from the unstructured text are hard to extract because of its multifaceted and individualistic characteristics of it. Traditional natural language processing techniques for entity and relation extraction fail for such entity-specific properties. As a specific example, we proposed *a method for extracting properties describing human attributes in textual modalities.* While our evaluation covers specific property identifiers, FemmIR is generic and works for any class of identifiers, as long as the corresponding properties are available.

Limitations of previous works. Correlation learning methods [52–55, 99–101] linearly or non-linearly projects low-level features from representation models to a common subspace. Metric learning methods [102–104] learn a distance function over data objects based on a loss function to map them into the common subspace. All these models require a large amount of training data and data representations lack a common encoding mechanism. FemmIR closely relates to metric learning methods. Contrary to them, we do not directly correlate class labels or weak labels to the loss function. The proposed Edit distance between property graphs implicitly captures the signal for relevance.

In contrast to common representation learning models, *data discovery* models based on relational queries allows more flexibility to consider explicit information need from users, and use high-level properties in the system. EARS [1] is one such content-based data discovery system that, similar to our approach, takes user examples as queries and delivers relevant multi-media results. However, the prime aspect of EARS is it assumes a schema mapping among all modalities, and to introduce new modalities the common schema needs to be updated. In contrast, FemmIR offers a general solution to include retrieval from novel modalities for a diverse set of systems.

4.1.1 Preliminaries & Problem Definition

In this section, we first provide formal definition to attributed relational graph. We then proceed to formulate the problem of multi-modal information retrieval for property-specific information need.

Definition 4.1.1 (Attributed relational graph)**.** *An attributed relational graph (ARG) is a graph whose nodes and edges have assigned attributes (single values or vectors of values from* Σ*). For the sake of simplicity, from now on we denote the node and edge attributes by labels, as labels are specific type of attributes. Although we focus our methodology only on directed and labeled graphs, it is designed to handle any forms of graphs. It is defined as:* $g = (N, E, l)$ *where*

1. N *is the finite set of nodes,*
2. $E \subseteq N \times N$ *is the set of edges,*
3. $l : N(g) \cup E(g) \to \Sigma$ *is a labelling function that assigns each vertex and/or edge a label from* Σ*. Specifically,* $l(u)$ *and* $l(u, u')$ *are the label of node* u *and the label of edge* (u, u')*, respectively,*
4. Σ *is a finite or infinite set of unconstrained labels.* $A \in \Sigma$ *represents labels enumerating the node-type.*

Definition 4.1.2 (Graph Edit Distance)**.** *Formally, the edit distance between* g_1 *and* g_2*, denoted by GED* (g_1, g_2)*, is the number of edit operations in the optimal alignments that transform* g_1 *into* g_2*, where an edit operation on a graph* g *is an insertion or deletion of a node/edge or relabelling of a node/edge. Intuitively, if two graphs are identical (isomorphic), their GED is 0.*

Considering a collection of data from $\mathcal{M} \in \mathbb{Z}^{+}$ modalities, we denote the j-th sample of the i-th modality as $\mathbf{d}^{\mathrm{i}}_{\mathrm{j}}$. The set containing all the $n_{\mathrm{i}} \in \mathbb{Z}^{0+}$ samples of the i-th modality is denoted as $\mathcal{D}_{\mathrm{i}} = \{\mathbf{d}^{\mathrm{i}}_{1}, \mathbf{d}^{\mathrm{i}}_{2}, \ldots, \mathbf{d}^{\mathrm{i}}_{n_{\mathrm{i}}}\}$. Each data sample contains a collection of *object-properties.* For example, a document has a *topic, metadata*, and *entities* with their *relationships*, along with any *event* it describes. Let $\mathcal{O} = \{\mathbf{o}_1, \mathbf{o}_2, \ldots, \mathbf{o}_l\}$ be the set of all such *object-properties*, where z_r is the set of values of property $\mathbf{o}_r$. A data sample $\mathbf{d}^{\mathrm{i}}_{\mathrm{j}}$ is described with a subset of $\mathcal{O}$. $\mathcal{O}_E \subseteq \mathcal{O}$ denotes the set of object-properties describing an entity E. $z_r = \{\phi\}$ indicates that $\mathbf{o}_r$ is not present in $\mathbf{d}^{\mathrm{i}}_{\mathrm{j}}$. Property identifiers implement a relation, $PROP$ $(\mathbf{d}^{\mathrm{i}}_{\mathrm{j}}) \subset \mathcal{O}$ that maps a data-sample to a set of object-properties ($PROP$: $\mathcal{D} \to \mathcal{O}$). A query is issued against a corpus with $\mathcal{M}$-modalities, $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_{\mathcal{M}}\}$.

Problem 4.1 (Multimodal Information Retrieval)**.** $Q_{\mathbf{d}^{m\in\mathcal{M}}} = \{\mathbf{o}_1 = z_1, \mathbf{o}_2 = z_2, \ldots, \mathbf{o}_p = z_p\}$ *is a data query that is expressed in one of the two ways:*

(1) (Query-by-Properties) with p object-properties mentioning a target data-sample $\mathbf{d}^m_q$ *from modality m with PROP* $(\mathbf{d}^m_q) = Q_{\mathbf{d}^{m\in\mathcal{M}}}$*, or*

(2) (Query-by-Example) with an example data-sample $(\mathbf{d}^m_q)$ *of modality m with PROP* $(\mathbf{d}^m_q) = Q_{\mathbf{d}^{m\in\mathcal{M}}}$*.*

The task is to retrieve a ranked list, $R = (\mathbf{d}^{x_1}_1, \mathbf{d}^{x_2}_2, \ldots \mathbf{d}^{x_t}_t)$ *of* $t \in \mathbb{N}_0$ *data-samples from all available modalities in the system satisfying PROP* $(\mathbf{d}^m_q)$*, where* $\mathbf{d}^{x_c}_c$ *is c-th data in R from modality* $x_c \in_R \mathcal{M}$*.*

Relevance is scored based on the degree of common object-properties between the data-object $\mathbf{d}^{x_c}_c$ in the ranked list, and the query data $\mathbf{d}^m_q$, $PROP$ $(\mathbf{d}^m_q) \cap PROP$ $(\mathbf{d}^{x_c}_c)$. A similarity score is used to define the degree of relevance, $0 \leq SIM$ $(\mathbf{d}^{x_c}_c,\ \mathbf{d}^m_q) \leq 1$. Similarity score of 0 indicates non-relevance, whereas a score of 1 indicates complete relevance and a proper subset, $PROP$ $(\mathbf{d}^m_q) \subset PROP$ $(\mathbf{d}^{x_c}_c)$.

Our problem setting assumes that the user has knowledge about o_p and their corresponding z_p for Query-by-Properties. This assumption is realistic in real-world scenarios and has been considered in multimodal data query literature where properties are used to express the information need [1, 96, 97].

4.2 Multimodal Information Retrieval

We will now describe the multimodal similarity matching method to find the relevant data to user provided information need (mentioned with an example, or with object properties).

The matching algorithm considers the data samples, $\mathbf{d}_c^{x_c}$ (from the data repository, or from data streams) and user provided example, $\mathbf{d}_q^m$ as input and outputs the similarity score between them: $SIM\ (\mathbf{d}_c^{x_c},\ \mathbf{d}_q^m)$. The corresponding object-properties are assumed to be available from each data sample extracted by the system-specific property identifiers before the matching algorithm is applied. We propose a weakly supervised approach to rank the data samples by generating a distance metric between them based on the amount of edits (changes) needed to convert the properties of one sample to another instead of manually annotating the number of matched object-properties. To this end, we first process the input data samples with a graph ingestion mechanism which converts the extracted properties into a hierarchical attributed relational graph (HARG). Our weakly supervised strategy, FemmIR adopted the Munkers' algorithm [88] to calculate the edit distance between the data samples. Finally, we used a neural network based edit distance approximation algorithm to learn a function to map the graph embedding of the HARGs to a similarity score between the data samples. During inference, the model just takes the extracted properties from the data samples, and outputs the similarity score by using the mapping functions. We start with an example scenario to demonstrate how data ingestion works and then proceed to describe the weakly supervised approach.

4.2.1 Data Ingestion with Graphs

Consider the task of finding the location of a person from large amount of video data using the text queries or reports (Example 4.1). The system finds the video feeds that has the persons similar to the report description (using multi-modal similarity matching) by focusing on object-properties of persons in the video and text. The goal is to identify the similarity score between video feeds, text queries, and incident reports which can be used to deliver a ranked list of relevant data samples to the user.

Observations. We make the following observations.

O4.2.1 The number of object-properties that is used to compare between two data samples are finite, and the value of the properties are mostly categorical values. A data sample can describe a large amount of objects and object-properties, but for

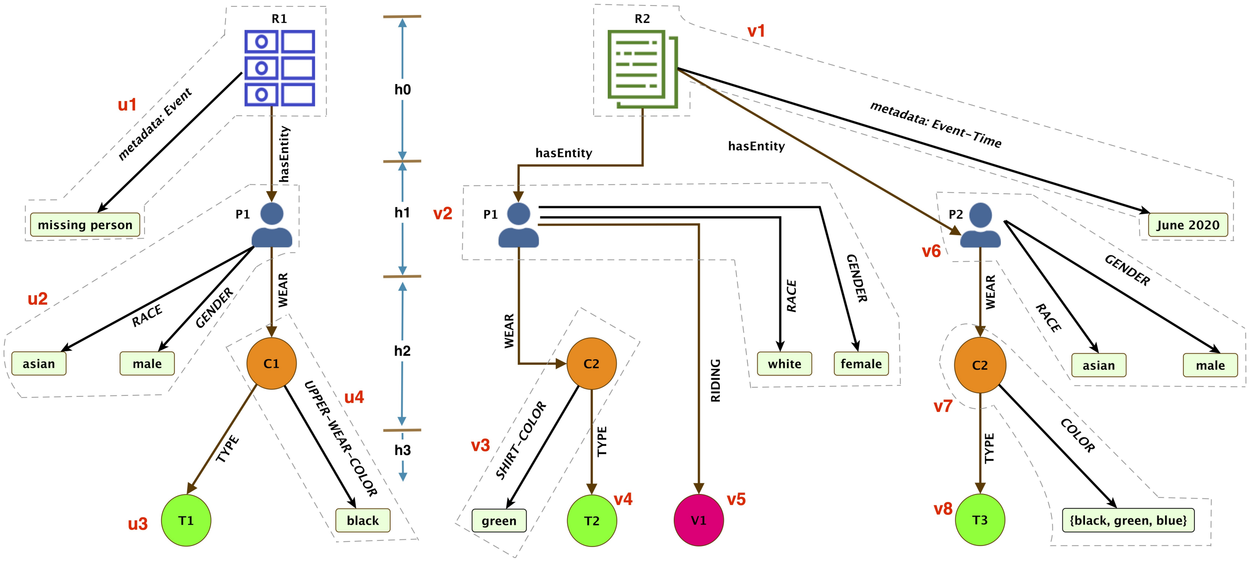


Figure 4.1. HARG and Weak Label Generation; Left side graph refers to g^q, and the right side graph refers to g^c. Node-type labels are as follows. V: EPL Vertex, R: Root, P: Person, C: Clothes, T: Type, M: Motor-Vehicles. Squared nodes correspond to the non-empty leaf nodes.

system-specific similarity comparison an user is only interested in a finite number of properties.

O4.2.2 Data samples are objects themselves that have different properties such as, meta-data, topics, and events that they describe. *Entities* are specific types of objects described in a data sample which has its own properties.

O4.2.3 *Relationships between objects* are a specific types of object-properties which belong to all participating objects. The set of values corresponding to the objects would be complementary to each other. Value for relation-name can be different for the same relationship through different data samples. For example, different text would describe the same action in different forms: *wearing, wear, has.*

O4.2.4 Some properties in z_p have single and fixed value-set i.e., GENDER, RACE, HEIGHT, while other properties have a multiple number of values in their value-set i.e., CLOTHES.

O4.2.5 Some object-properties such as, CLOTHES-COLOR have different values for different data samples. For example, in Figure 4.1, `UPPER-WEAR-COLOR, SHIRT-COLOR, COLOR` all refer to the color of clothes.

Our intuition here is that entities, relationships, and object-properties in a data sample have a inter-connected structure and if we can capture the changes we need to make to this structure to convert it to structure of another data-sample, then we can capture the differences between these samples. Based on this intuition, FemmIR starts by constructing a *hierarchical attributed relational graph*, called (HARG), with a common hierarchy for all data samples. The choice of graph as a representation was influenced by:

1. graph being the best data structure to capture information from connected structures,
2. based on observations **O4.2.3** and **O4.2.5**, a data structure with representation-invariant encoding mechanisms that can capture the syntactic similarities between different values was necessary.

Definition 4.2.1 (Hierarchical Attributed Relational Graph)**.** *HARG is a specific type of ARG in the form of a multi-level tree with* `heve`*s. It consists of a root node, multiple levels of nodes and edges emanating from it, and specific type of leaf nodes. Nodes at level h is denoted by N^h.*

CONSTRUCT-HARG. Each data sample is represented as HARG, following the steps:

1. The graph starts with a single node at level 0 ($h = 0$) containing a common-label (CONTENT/ OBJECT/ ROOT) for all data samples in the same application domain: $l(N^0) = \{ROOT\}$.
2. Level 1 nodes constitute of the object-properties of the data sample itself where the property-name is the edge label, and the property-value is the node label: $l(N^0, N^1) = \mathbf{o}_p, l(N^1) = z_p$.

   With the exception of o_p being an entity of that data sample, N^1 would be a leaf node. And for entities, we define the edge label as $l(N^0, N^1) = \{hasEntity\}$.
3. In case a set of $\mathbf{o}_p$ describes the object-properties of an entity, $N^k (k \geq 1)$ will be a pointer to the properties of that entity, whereas $l(N^k) = \{entity\text{-}type\}$. We categorize entities

in two groups for each data sample: *primary*, and *secondary*. Level 1 of HARG only contains primary entities.

4. Level 2 and subsequent levels contain the property-values of the entities in the previous level with $l(N^k, N^{k+1}) = \mathbf{o}_p(k \geq 1)$ and $l(N^k) = z_p(k \geq 2)$. From Definition 4.2.3, for RELATION properties, $\langle R, S, Arg \rangle$ where entity-pointer S is at level-k and entity-pointer Arg is at level-$(k+1)$, $l(N^k, N^{k+1}) = R, l(N^k) = S, l(N^{k+1}) = Arg$.

5. There can be edges between entities in the same level with RELATION properties, R. With nodes N^k and N^r,
$l(N^k, N^r) = R$, where $l(N^k) \neq l(N^r)$ but $k = r$.

6. The leaf nodes of HARG always contain a property-value or a NULL value for $z_p = \{\phi\}$.

Definition 4.2.2 (Primary Entities)**.** *Primary entities are entities that take the role of a subject in terms of a verb.*

1. *In visual modalities, entities that control the action or relation properties are considered as primary entities.*

2. *Entities that satisfy any of the following criteria is considered as primary entities in textual modalities:*

   (a) *In phase structure grammars, primary entity is an immediate dependent of the root node [105],*

   (b) *In dependency grammars, primary entity is an immediate dependent of the finite verb [106].*

3. *For database records, we consider the entities from the tables with no foreign key constraints as the primary entity.*

Secondary entities include any entities not satisfying the conditions of primary entities including objects, verb arguments, and themes.

Definition 4.2.3 (Relation between Objects)**.** *Object-properties describing a relationship or action R between two entities S (initiator) and Arg (outcome/ receiver/ modifier) are defined as* RELATION *properties, and the property-value is defined as a triplet of $\langle R, S, Arg\rangle$. For a n-ary relationship R, identifiers associate each action with multiple entity arguments, Arg_1, Arg_2, $\ldots$, Arg_{i}, $\ldots$, Arg_n with* role R_o^{i}*. n-ary relationships are broken into multiple binary relationships with $l(N^k, N^{k+1}) = \{R : R_o^{\mathrm{i}}\}, l(N^k) = S, l(N^{k+1}) = \{Arg_{\mathrm{i}}\}$.*

Figure 4.1 demonstrates two example hierarchical attributed relational graphs from the experimental dataset. $R1$ and $R2$ refers to two different data samples. For the leaf nodes $T2, M1$, and $T3$ in $R2$, $z_p = \{\phi\}$. `Wear`, and `riding` refers to the Relation property, where `Persons` are subjects, and `Clothes` and `Motor-vehicles` are arguments. We made two assumptions for the generation process:

(**I**) We assume prior knowledge of the system-specific properties [ ] and that they have been extracted with appropriate property-identifiers,

(**II**) The entity types for node labels are system-specific, and must be consistent through lifetime of the system. This assumption is valid since the property identifiers from each modality would be system-specific and extracted object types would be consistent across data samples.

4.2.2 Weak Label Generation

FemmIR further defines a new distance metric, Content Edit Distance (CED) using a variation of the Munkres' algorithm [88] to calculate the amount of edits (changes) for optimal alignment of the query-example HARG to HARG of another data-sample. CED is considered as weak label for the retrieval task for two reasons:

1. Munkres' algorithm is suboptimal as it only calculates approximate edit distance values,

2. the quality of HARG rely on the choice of primary entity selection which can be noisy.

Our intuition was graph edit distance (GED) calculation algorithms (A*-search, VJ, or Beam) would be enough to calculate the number of changes after we have build the HARGs, but we made following observations.

O4.2.6 Different nodes and edges in HARG have different change cost. User should be allowed to specify individual property replacement cost.

O4.2.7 GED calculation algorithms differ in speed based on the number on nodes and HARG contains variable sized graphs.

O4.2.8 Object-properties such as, RELATION has dependency between different levels of HARG and should not be considered individually during the change estimation. For example, for *person wearing clothes*, edit cost for `person` and `cloth` should be considered together between different data-samples.

O4.2.9 Considering **O4.2.4**, we cannot calculate the edit cost of certain properties just by replacing or deleting them since they have multiple number of values in their value-set.

For properties with list values, we consider two types of comparison: **(propertyLED)** ordered comparison with modified *Levenshtein distance*, and **(hashComp)** unordered comparison with *hash table*. Summing the cost of edits for all the properties between two data-samples ignores the inter-connected structure among the properties. In Figure 4.1, the graph from $R2$ has two persons, and while comparing with $R1$ we would want to know the minimal edit cost by considering which person in $R2$ is closer to the person described in $R1$. Content Edit Distance calculates the cost for the minimal cost alignment of one data-sample to another. Since only property values in leaf nodes in a HARG have direct replacement cost, we propose a new kind of vertex in HARG, *Entity-with-Property-in-Leaf (EPL) vertex* (Definition 4.2.4) for calculating the cost for an individual object assignment. Given $EPL(V)$ is the finite set of EPL Vertices, $EPL(E) \subseteq EPL(V) \times EPL(V)$ is the set of edges, and $EPL(l) \subset l$ is the labeling function, a HARG is now defined as:

$$g_{\text{epl}} = (EPL(V), EPL(E), EPL(l))$$

Definition 4.2.4 (Entity-with-Property-in-Leaf Vertices)**.** *A node labeled with object-type (A) with their outgoing edges labeled with object-properties ($\mathbf{o}_p$) and the connected leaf nodes labeled with property-values (z_p) are considered as* Entity-with-Property-in-Leaf *(**EPL**) Vertex, $EPL(V)$. A node without any leaf nodes is also considered as an EPL vertex. An EPL vertex can be connected to other EPL vertices and have their own cost functions.*

Munkres Algorithm for CED calculation. We consider the CED calculation as an assignment problem and adopted the bipartite graph matching method in [88]. Compared to the exponential time-complexity of A*-search, Munkres' [88] algorithm has a polynomial time complexity. Estimating content edit distance instead of a simple property-to-property comparison allows the flexibility to consider the dependency between properties and graph levels. Given the non-empty HAR graph from query-example, $g^q_{\text{epl}} = (EPL(V)^q, EPL(E)^q, EPL(l)^q)$ and the HAR graph from the compared data-sample, $g^c_{\text{epl}} = (EPL(V)^c, EPL(E)^c, EPL(l)^c)$, where
$EPL(V)^q = \{u_1, \ldots, u_n\}, EPL(V)^c = \{v_1, \ldots, v_m\}$, the Munkres' algorithm would output CED (g^q_{epl}, g^c_{epl}). We made the following adjustments to the Munkres' algorithm in [88].

1. EPL-vertices in the query graph needs to be aligned to the data-samples, hence we will fix the assignment size k to $|EPL(V)^q|$.

2. For data retrieval, the entities and relations in query graph needs to be in comparison-graph, otherwise indicates missing property. So there is no need to add dummy nodes to g^q_{epl}. Formally, if $n > m$, only the costs for $max\{0, m - n\}$ node insertions have to be added to the minimum-cost node assignment.

3. Next, the $n \times m$ cost-matrix C is generated. (1) For different type of objects A in u_{i} and v_{j} the replacement cost is set to ∞. (2) The cost for a single object assignment $C_{\text{i,j}}$ is calculated by comparing the property values z_p (normal-comparison and list-comparison) in EPL-vertex u_{i} and v_{j}.

4. To accommodate for **O4.2.5**, while applying Adjacency-Munkres, we set the default cost of an edge replacement $c(\text{e}_{u_{\text{i}}} \rightarrow \text{e}_{v_{\text{j}}})$ based on the Wu-Palmer distance between Synsets of

$l(\mathrm{e}_{u_\mathrm{i}})$ and $l(\mathrm{e}_{v_\mathrm{j}})$. $\mathrm{e}_{u_\mathrm{i}}$ denotes all edges connected to u_i and $\mathrm{e}_{v_\mathrm{j}}$ denotes all edges connected to v_j. In general, any language embedding can be used instead of Synsets.

$$c(\mathrm{e}_{u_\mathrm{i}} \rightarrow \mathrm{e}_{v_\mathrm{j}}) = 1/wpdist(s_{l(\mathrm{e}_{u_\mathrm{i}})},\ s_{l(\mathrm{e}_{v_\mathrm{j}})}) \tag{4.1}$$

Cumulative-Munkres. Using Adjacency-Munkres from [88] allows us to find the optimal assignment of each EPL vertex without taking into account the dependency among them **O4.2.8**. We utilize the levels from HARG to include the dependency information into the cost-matrix. So for every $C_{\mathrm{i,j}}$ in the cost matrix from adjacency-munkres denoting an assignment of u_i to v_j, we add their parent EPL-vertices assignment cost to $C_{\mathrm{i,j}}$, starting from EPL-vertices in level-1. In the remainder of this paper, we will call this method Cumulative-Munkres since it uses the cumulative cost of the parent and child nodes to preserve the dependency information.

4.2.3 Similarity Measurement

Finally, we propose to use an end-to-end neural network model, SimGNN [89] to learn an embedding function to map d_q and d_c into a similarity score based on the CED score. User requirements (such as, relationships between properties, searching in a time range, or within a specified location, etc.) and system constraints (such as, different property-values) are applied with appropriate replacement costs while calculating CED. Similarity scores for training the model are derived by transforming the distance scores using the normalization method from [107] and an exponential function on the normalized score. (Line 37 in Algorithm 3). The embedding function outputs a number of interaction scores between a pair of graphs using Neural Tensor Networks (NTN) [108] on the graph embeddings. For calculating the graph embedding, first, Graph Convolutional Networks (GCN) [109] are used on the HARG to obtain the node embeddings. GCN is representation-invariant and allows us to account for different kinds of labels for nodes and edges, when ground truths are available. It is also inductive and allows to compute the node embedding for any unseen graph following the GCN operation, which makes it a great choice for variable sized FemmIR graphs. Then, an attention network is used to combine the node embeddings into a graph embedding allowing to learn each node's weight in the similarity determination as part of the end-to-end network. In addition, SimGNN augments the graph level interaction score with local information by calculating histogram features from a pairwise node interaction score between the node embeddings. Finally, a multi-layer fully connected network is applied to learn a single similarity score from the interaction scores, which is compared against the similarities from the weak-labels or the ground-truths using mean squared error loss.

$$C = \frac{1}{|D|} \sum_{d_c \in D} (\hat{s} - s(d_q, d_c))^2 \tag{4.2}$$

where D is the set of data samples from the repository or the stream, $\hat{s}$ is the predicted similarity score, and $s(d_q, d_c)$ is the ground-truth similarity between d_q and d_c. This similarity score allows us to rank the data samples against the query example.

4.2.4 FemmIR algorithm

Algorithm 3 presents the pseudocode of our retrieval algorithm FemmIR which takes two data samples as input and returns the similarity score between them as output.

Line 1 Extract the set of properties and their values, $\mathcal{O}^{\mathrm{j}}$ from data-sample d_{j} using the modality-specific property-identifiers.

Lines 2 - 3 Construct the Hierarchical Attributed Relational Graphs using the identified properties following the steps in Section 4.2.1.

Lines 4 - 37 During training, generate the CED as weak label using the Munkres algorithm. CED is used to calculate the similarity score, and this pair of data-samples and the similarity score is added as training sample for SIMGNN.

Line 5 Discover the EPL-vertices in the HARGs, and define g_{epl}.

Line 6 Initialize an empty $n \times m$ cost-matrix C.

Lines 7 - 8 Iterate through all the vertices in $EPL(V)^q$ and $EPL(V)^c$ and compare the properties in each vertex to assign the costs.

Line **9** For different types of object, set the cost to ∞, not allowing different types of object to be aligned.

Line **13** If a property in u_{i} is absent in v_{j}, it needs to be inserted in v_{j}. Increment the cost-matrix value by the insertion-cost.

Lines **15** **-** **19** If the property is not a list, then just compare the values in u_{i} and v_{j}. If they mismatch, add the replacement cost to the cost-matrix, otherwise nothing is added.

Lines **21** **-** **22** If the property is a list, we need to compare them either with a Levenshtein distance (ordered comparison) or with a hashmap (unordered comparison) from Section 4.2.2. cmpis a control variable to specify what kind of comparison is required. The overall cost is added to cost-matrix.

Line **25** For applying Adjacency-Munkres, the minimum edge replacement cost is added to the cost matrix using Equation 4.1.

Lines **28** **-** **32** If Cumulative-Munkres is required (set by mType), cost-matrix entry of the parent vertices are added to each $C_{\mathrm{i,j}}$.

Line **36** Apply the Munkres algorithm to calculate the optimal assignment based on C, and the associated cost is the CED.

Line **37** Normalize CEDto the graph sizes, and apply an exponential function to convert it to a similarity score in the range of (0, 1].

Lines **38** **-** **39** If in testing phase, apply the learned mapping function, SIMGNN to predict the similarity score from the HARGs.

3 Algorithm 3: FemmIR(d_q, d_c)

```
$\mathcal{O}^q \leftarrow \mathsf{PROP}(d_q), \mathcal{O}^c \leftarrow \mathsf{PROP}(d_c)$
$g^q \leftarrow \mathsf{constructHARG}(\mathcal{O}^q)$
$g^c \leftarrow \mathsf{constructHARG}(\mathcal{O}^c)$
if training then
   $g^q_{\text{epl}}, g^c_{\text{epl}} \leftarrow \mathsf{discoverEPLV}(g^q,\ g^c)$
   $C \leftarrow \phi$
   for $u_{\text{i}} \in EPL(V)^q$ do
      for $v_{\text{j}} \in EPL(V)^c$ do
         if $\mathsf{TYPE}(u_{\text{i}}) \neq \mathsf{TYPE}(v_{\text{j}})$ then
            $C_{\text{i,j}} = \infty$
         fi
         for $\mathbf{o}_p \in u_{\text{i}}$ do
            if $\mathbf{o}_p \notin v_{\text{j}}$ then
               $C_{\text{i,j}}$ += $\mathsf{icost}(\mathbf{o_p})$
            elseif $\mathsf{TYPE}(z_p)$ is not list then
                  // $z_p(u_{\text{i}})$ is value of $\mathbf{o}_p$ in vertex $u_{\text{i}}$
               if $z_p(u_{\text{i}}) \neq z_p(v_{\text{j}})$ then
                  $C_{\text{i,j}}$ += $\mathsf{rcost}(\mathbf{o_p})$
               else $C_{\text{i,j}}$ += 0
               fi
            else
               $C_{\text{i,j}}$ += $cmp * \mathsf{propertyLED}(z_p(u_{\text{i}}), z_p(v_{\text{j}}))$ +
               $(1 - cmp) * \mathsf{hashComp}(z_p(u_{\text{i}}), z_p(v_{\text{j}}))$
            fi
         endfor
```

```
3 Algorithm 3: continued
            $C_{i,j} = C_{i,j} + min\{\sum c(e_{u_i} \rightarrow e_{v_j})\}$
        endfor
    endfor
    if $mType$ then
        for $u_i \in EPL(V)^q$ do
            for $v_j \in EPL(V)^c$ do
                $u_{\hat{i}} = parent(u_i), v_{\hat{j}} = parent(v_j)$
                $C_{i,j} = C_{i,j} + C_{\hat{i},\hat{j}}$
            endfor
        endfor
    fi
    $CED(g^q, g^c) = Munkres(C)$
    $nCED = \frac{CED(\ g^q, g^c\ )}{(|g^q| + |g^c|)/2}$
    $SIM(d_q, d_c) = e^{-nCED}$
else
    $SIM(d_q, d_c) = \mathsf{SIMGNN}(\ g^q, g^c\ )$
fi
return $SIM(d_q, d_c)$
```

Generalization.

1. Algorithm 3 assumes that the edge labels for level 0 is fixed to *hasEntity* and *metadata* with granularity (such as, *time*, *location*, etc.). These are flexible and can be set to any labels in FemmIR as long as it is consistent throughout the lifetime of the system.

2. Object-types are assumed to be system-specific, and can be variable across different systems and applications. FemmIR can handle any labels for entity-type since the

retrieval result does not depend on it. The comparison between properties are affected by it which remains valid as long as same heuristics is maintained for all modalities in a system.

3. FemmIR is capable of handling different replacement costs and insertion costs for properties in different application domains.

4. For the edge replacement cost, any language embedding will work as long as the objective function places semantically similar tokens closer to each other.

4.3 Experiments

4.3.1 Dataset

We adopt the **MARS** person re-identification dataset from [98] to benchmark the property identifiers in visual modalities. MARS consists of 20,478 tracklets from 1,261 people captured by six cameras. There are 16 properties that are labeled for each tracklet, among which we used - GENDER (MALE, FEMALE), 9 BOTTOM-WEAR COLORS, and 10 TOP-WEAR COLORS.

Using the above-mentioned datasets, we built the **(InciText + MARS)** dataset to evaluate the retrieval performance of FemmIR. The composition statistics for each modality are:

1. Image (3270/1100/1144),

2. Text (296/178/145), and

3. Video (1454/499/539),

where (*/*/*) stands for the sizes of training/validation/test subsets.

For the ground-truth, we ranked the data samples in ascending order of the penalties for the mismatched properties. The properties were chosen depending on the user requirement, and the mismatches were assigned different penalties. In Example 4.1, an officer searches in the following order: (1) same gender and race, (2) same bottom clothing, and (3) same top clothing. The intuition behind this is if there is a gender mismatch, they are definitely not the same person. It is possible for a person to change the top clothing in a short span of time, but it is harder to change the bottom clothing. So even if there is a mismatch on top color, there is a chance of it being the same person given similar time-span and vicinity. Therefore, we set the penalty for each mismatched property as follows: $rcost(\textsc{top-color}) = 1$, $rcost(\textsc{bottom-color}) = 2$, and $rcost(\textsc{gender}) = 3$, with gender having the highest penalty, hence the highest importance. Exact matches are the top most in the ranking with a zero penalty.

4.3.2 Settings

We follow the original train/test partition of MARS [98] dataset for benchmarking. For models in [110–112], we formed a training batch by randomly selecting 32 tracklets, and then by randomly sampling 6 frames from each tracklet. During testing, F frames of each tracklet are randomly split into $\lfloor \frac{F}{n} \rfloor$ groups, and the final result is the average prediction result among these groups. We used a validation set of mutually exclusive 125 people selected from the training set. For color sampling, we used the result from the first frame from each tracklet. We compared three properties across all models - GENDER, TOP COLOR and BOTTOM COLOR. For the retrieval model, we only considered the synthetically generated part of InciText. For munkres, we used the API from clapper[1]. We did not use the local node-node interaction information during the training phase for FemmIR.

4.3.3 Benchmarking for Property Identifiers in Visual Modality

Since MARS has a large amount of ground truths for person attributes, we compared existing models from *Person Re-Identification* task. From the CNN models, we used the image-based Resnet50 [113] as baseline. Due to the temporal nature of videos, we also compared the 3D-CNN [112], CNN-RNN [111], Temporal Pooling and Temoporal Attention [110] models. As a heuristic based model, we chose the color-sampling model from [35]. Figure 4.1 describes the bench-marking results for the compared models. Resnet50 performed significantly better than other models for bottom-color, while temporal attention worked

[1] ↑clapper

best for top-color. Considering the average performance on all attributes, we choose the *image-based CNN model* for the retrieval task. Since the properties in our task are all motion-irrelevant, the video based extraction models do not have a large impact on the performance. In terms of training data and time, color sampling surely has an advantage. Resnet50 needed 513 minutes and the temporal attention model needed 1073 minutes for training, whereas color sampling has zero training time. Color sampling works by isolating body regions and evaluating on pixel values, hence the presence of sunlight or clouds may have adversely affected the performance.

4.3.4 Retrieval Performance of FemmIR

We compared FemmIR with the EARS method [1]. Since EARS does not require any training, we only used the test set in (InciText + MARS) . We formulated the JOIN queries in EARS method on properties from Example 4.1. The results were a union among an exact match, and partial matches for the individual properties.

For evaluation, we considered cross-modal retrieval tasks as retrieving one modality by querying from another modality, such as retrieving text by video query (Video Text) or, retrieving image by text query (Text Image). We also show the comparison for multi-modality retrieval. By submitting a query example of any media type, the results of all media types will be retrieved such as (Image All, Text All). We adopt mean average precision (mAP) as the evaluation metric, which is calculated on all returned results for a comprehensive evaluation. We consider data samples with CED < 3 in comparison to the query object, as ***relevant*** for that query. This would return contents where persons only with color mismatches are found.

With an average F1 score of 79.59% for video and image property identifiers, the mAP scores of image and video queries are 27%-37%. Text modalities with their high-performance identifiers get the highest mAP across modalities. This indicates the dependence on property identifier performance. Precision-recall (PR) curves in Figure 4.2a and 4.2b show that at lower degrees FemmIR perform comparably with EARS, but with higher degrees of recall, the performance degrades. We will perform ablation studies (using local node-node interaction or eliminating imbalance of modalities in training data) in the future for FemmIR performance improvement.

4.4 Related Works

4.4.1 Metric Learning.

[114, 115] uses hinge rank loss to minimize intraclass variation while maximizing interclass variation. [102] minimized the loss function using hard negatives with a variant triplet sampling, but needs fine-tuning and augmented data. [103] uses an additional regularization in the loss function with adversarial learning. [104] enables different weighting on positive and negative pairs with a polynomial loss function. FemmIR has similarities to metric learning with the objective of minimizing the edit distance between two graphs. In contrast, FemmIR re-uses pre-extracted properties and does not require data samples to create positive-negative pairs.

4.4.2 Weakly Supervised Learning.

[85, 87] use weak signals from entity and relationship similarities retrieved from video captions and text. [1] assumes knowledge of the translation module which makes it less adaptable to novel modalities. [86] uses a similarity-based retrieval technique to extract images with similar subsurface structures. FemmIR also uses a weak signal approach for ranking relevant samples from multiple modalities, but the weak labels are constrained to use the pre-extracted properties and must implicitly maintain the structure between the entities and relationships.

4.4.3 Semantic Understanding with Encoding Networks.

[56, 57, 116, 117] learns semantically enriched representations of multi-modal instances by using global and local attention networks. Similarly, FemmIR uses graph convolutional network [109] to align the most important nodes contributing to the overall similarity, denoting the most similar properties between samples.

4.4.4 Content-based Data Discovery.

[1, 60, 70, 96, 97, 118] implement content-based data retrieval by taking user-provided example records as input and returning relevant records that satisfy the user intent. Our work shares similarities to DICE [70], which finds relevant results by finding join paths across tables within the data source. However, it focuses on discovering relevant SQL queries from user examples, whereas FemmIR focuses on finding the relevant content directly by finding similar object properties. EARS [1] finds relevant data by applying JOIN queries on the user-required properties from different modalities. Similar to EARS, we also assume the knowledge of pre-identified properties. EARS can scale to petabytes of data, but it needs additional queries to retrieve soft similarities. The number of SQL queries increases proportionally to the number of properties in the user query. Contrary to EARS, we do not assume a common schema for all modalities and do not require re-training from scratch to accommodate new modalities.

4.4.5 Cross-modal Correlation Learning.

[52, 53, 99, 119] use canonical correlation learning to linearly project the low-level features into a common subspace. For non-linear projections, [54, 55, 100, 101, 120] extended the linear methods [55] or used shallow [54, 101] or deep networks [100] to learn the correlations. SDML [121] removes the dependency of jointly learning from all modalities by predefining a common subspace and using a deep supervised auto-encoder for each modality. DSRL [59] directly learns the pairwise similarities by integrating relation learning, capturing the implicit nonlinear distance metric which FemmIR also focuses on. Most of these works assume the presence of class labels [53, 119], choice of appropriate feature extraction, and translation models for specific modalities. This limits the capability to integrate new sources or use pre-existing features/properties. FemmIR separates the feature extraction modules from the retrieval module and integrates pre-identifier property from any modality using graph encoding networks.

4.5 Summary and Future Directions

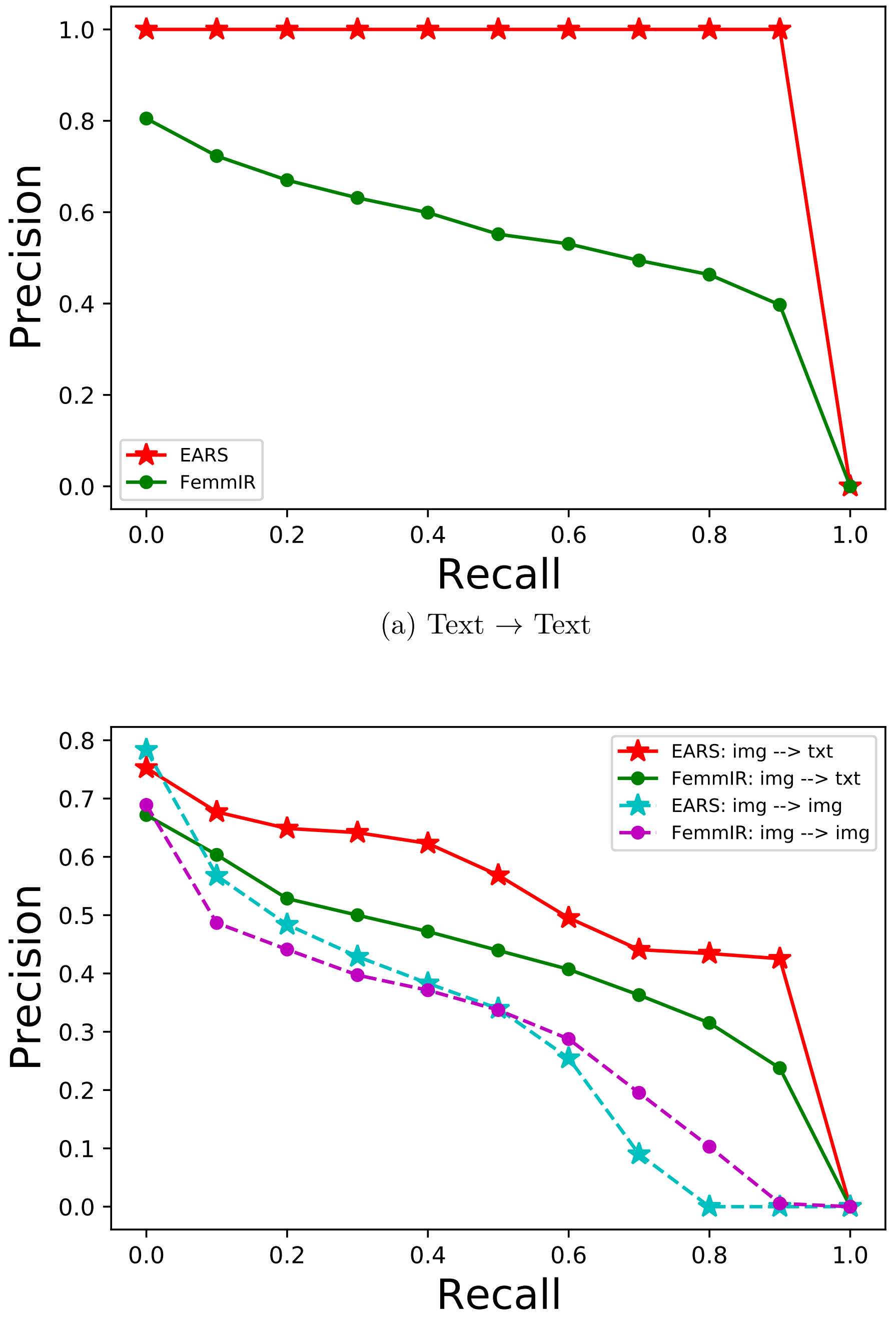
We introduced the problem of mismatch between the information need and model features, along with the lack of annotated data for multi-modal relevance. To this end, we presented FemmIR, a framework that uses weak supervision from a novel distance metric for data objects, and uses explicitly mentioned information needs with existing system-identified properties. We demonstrated the performance of FemmIR in identifying the relevant data to the user example without supervised training and additional computational resources. FemmIR has successfully implemented a *missing person* use case and is being updated to provide further assistance to local agencies in social causes. In the future, we plan to extend FemmIR to include multi-objective and evolving information needs to support more real-world use cases.

Table 4.1. Comparison of Property Identifiers for Videos with Accuracy (acc) and F1 measure on MARS dataset (%)

Properties	CNN (Resnet50)		3D-CNN		CNN-RNN		Temporal Pooling		Temporal Attention		Color Sampling	
	acc	F1	acc	F1	acc	F1	acc	F1	acc	F1	acc	F1
top color	75.22	73.98	67.91	65.19	70.54	67.33	74.98	73.13	76.05	74.64	44.65	38.31
bottom color	73.55	54.09	59.77	36.56	67.71	44.44	71.69	47.84	70.15	46.89	45.26	15.88
gender	90.01	89.71	86.49	76.22	90.07	89.62	91.04	90.63	91.82	91.48	-	-
average	**79.59**	**72.59**	67.97	59.18	76.11	67.13	79.24	70.53	79.34	71.01	44.96	27.10

Table 4.2. Performance of EARS and FemmIR in mAP(%)

Query	Target	EARS	FemmIR
Image	Text	0.54	0.40
	Image	0.27	0.27
	Video	0.33	0.29
	All	0.30	0.28
Text	Text	1.0	0.52
	Image	0.37	0.29
	Video	0.46	0.33
	All	0.43	0.31
Video	Text	0.62	0.43
	Image	0.30	0.29
	Video	0.37	0.30
	All	0.34	0.30
	Avg	**0.44**	**0.33**



(a) Text → Text

(b) Image → Text, Image

Figure 4.2. Performance of FemmIR on (InciText + MARS) .

5. WEAKLY SUPERVISED JOINT EMBEDDING FOR MULTI-MODAL INFORMATION RETRIEVAL (WeSJem)

We present *WeSJem*, a weakly supervised open-learning framework for jointly learning data representations from all modalities in a shared low dimensional vector space, by exploring the structural components of the data samples. The framework characterizes and formulates responses to different novelties encountered during multimodal retrieval from unknown application domains, user requirements, or temporal changes. *WeSJem* follows a three-step process: (1) Different modalities of data are translated to textual descriptions. (2) Weak similarity labels are generated among data samples by comparing topics and different structural elements (entities, relationships, and events) of the text. (3) Vector representations are learned for the data samples in the joint embedding space by exploring the relationships among the topics and structural elements. We address the supervision bottleneck problem, and show that topics and structural features can be used as a weak-supervision source, as well as provide a better semantic representation for retrieval of similar multi-modal data. Initial experiments are conducted using documents and videos as multi-modal sources, and topic as weak labels. In comparison to unsupervised methods, LSI and LDA, our model showed promising performance to capture the similarities in the low dimensional space.

5.1 Introduction

Finding relevant data from large data sources is the pre-requisite for any data analysis task. Current data discovery systems require human hours to sift through the large influx of multi-media data (e.g., text, image, video, audio, and 3-D model) for data preparation task. Multi-modal information retrieval takes queries in one modality to retrieve relevant data from other modalities, augmenting information from a single source with information from other sources. Cross-modal retrieval results can be improved if context is introduced in learning the relevance. For example, uploading the image of a person in google search returns images of similar cloths that the person is wearing. User experience would improve if the search results include the images of similar cloths, videos of people wearing similar cloths, or places where these kind of cloth can be purchased.

Previous works on multi-modal information retrieval have followed the idea of projecting modality-specific features from different modalities into a shared embedding space. [53–55, 99, 100, 119, 120] focuses on correlation learning to learn the projection function, using both pairwise information and class labels. [121, 122] uses auto-encoders to find correlations. Metric learning methods [56, 102–104] learn a distance function over data objects based on a loss function. Attention mechanism [56], [123, 124] proposes pre-training models for better generalization. While the aforementioned learning methods exhibit good performance on benchmark datasets, they suffer from the lack of labeled training samples for data discovery in practice. In open world environment, test data distribution is almost always different from the training data distribution. Current works do not focus on the noise in the input data, or the data relevance change over time. Many of the learning methods focus only on uni-modal or bi-modal retrieval, and cannot generate results for queries of all media types. Moreover, most of the above models suffer from the lack of explainable reasoning on how two different multimedia data are similar.

We propose a **We**akly **S**upervised **J**oint **Em**bedding model (WeSJem) for multi-modal information retrieval. Our model adopts the metric learning approach [125] as the backbone, and proposes to build a data information network as the weak signal generator. It has four components, including a translation module, a weak label generator module, a data information network, and multi-task learning. In detail, we first generate a dense video caption from the videos using a proposal and captioning module. Then we learn the weak labels using existing single modal encoders and text feature extractors. The separate stream design allows scalability to very large datasets in retrieval tasks. Finally, we create a data information network among all different modalities in terms of their similar features. A multi-task joint objective performs on the network, which aims to learn better representation for each data sample while maintaining multiple degrees of similarity among them. The objective function aims to maintain the inter-connection between different data modalities based on their structural features. Even in the absence of the weak labels, the objectives can be adapted to be trained in an unsupervised setting. The translation module and the weak label generation module are independent of the embedding architecture and can be replaced.

5.2 Related Works

Correlation Learning. Traditional cross-modal retrieval models focus on *correlation learning* to project data instances into a latent common subspace. [99] implements linear projection using canonical correlation analysis to optimize only the pairwise information. [53] learns the common features using class labels as a linkage to model correlations. Joint representation learning [119] constructs graphs to jointly model the correlation and semantic information with sparse and graph regularization. For non-linear projection, deep Canonical Correlation Analysis (DCCA) [120] uses modality specific subnetworks. [55] extends DCCA with an auto-encoder regularization term. Multi-view Deep Network [100] uses a view-specific and a common sub-network to learn the common space. [54] overcome using only shallow networks for common stage with hierarchical networks.

Metric Learning. [114] proposes a deep coupled metric learning approach with two hierarchical non-linear transformations. [115] used a hinge rank loss as objective function to map visual and semantic features into the shared space. [102] minimized the loss function using hard negatives with a variant triplet sampling. [103] introduced an additional regularization in the loss function with a modality classifier as part of the adversarial learning. [104] enables different weighting on positive and negative pairs with an universal weighting framework and a polynomial loss function.

With recent advancements in encoder-decoder networks [126, 127], [123, 124] provides a solution of pre-training the model on a large scale dataset. [122] used correspondence autoencoders to find correlations between images and text. [121] removes the dependency of jointly learning from all modalities by predefining a common subspace. [59] avoids explicitly learning a common space by integrating relation learning. [61] computes modality specific similarities with neural tensor networks. [56] uses attention mechanism to align multimodal embeddings learned through a multimodal metric loss function. [128] describes a unified framework for formal theories of novelty in learning algorithms, which is applied towards different domains, including multi-agent game, and open world image recognition. [129] discusses a self initiated open world learning agent with the example of a conversational bot in a hotel. [130] discusses characterization and changes of environments in which a radically autonomous physical agent can operate.

Most of these models require annotated labels specifying which data samples belong to the same category. The novelty frameworks does not explain information retrieval as a domain. WeSJem has close resemblance to metric learning and intermediate fusion approaches. Our approach differs from existing works in terms of representation learning methodology and independent module flexibility. Like [117], we also use modality specific encoders for translation module. The main difference in our proposed metric learning approach lies in building the data information network, and using the structural features as weak labels. Our method does not require annotated labels and we choose the positive and negative pairs such that similarity in structure is maintained. This work has the capability to take both the data instance and data features as query. WeSJem is capable of not only encoding the ontological information, but also pairwise and semantic information, if available.

Discussed existing works on retrieval task assume similar training and testing data distribution, and do not reflect on the novelties encountered during test. Existing works on novelty theories have proposed well-established frameworks, but to the best of our knowledge, this is the first work to formalize novelties in a domain with heterogeneous training instances and user-system interdependence.

5.3 Methodology

5.3.1 Problem Formulation and Overview

Multi-modal information retrieval is defined as retrieving the results of all modalities by submitting a query of any modality. Existing works tackle the problem in two phases: cross-media feature learning, and similarity measurement. The main contribution of this work is in cross-media feature learning. Formally, we consider the problem of information retrieval from a dataset $\mathcal{D}$ with a collection of data from m modalities. We denote the j-th sample of the i-th modality as $\mathbf{x}_{\mathrm{j}}^{\mathrm{i}}$. Modalities can include text documents, tweets, video snippets, images, and others.

The main goal of this work is to jointly learn high-quality vector representations for individual data samples from unlabeled multi-modal data set. We design our embedding function $\mathcal{F}$ to map the multi-modal data samples into a low dimensional vector space, such that multiple degrees of similarity are preserved in the embedding space. During inference, the similarity between two projected data samples $sim(\mathcal{F}(\mathbf{x}_p^v), \mathcal{F}(\mathbf{x}_q^t))$ will be measured in the joint space, using the existing methods such as the Cosine or the Euclidian distance.

Our main insight is that representing data in terms of different structural features through which different modalities of data can be similar, can provide us with a source of weak supervision for cross-modal retrieval. Our motivation comes from how structural representation of a raw unstructured text allows readers to infer better knowledge. Structural representation of a document entails topics, entities, events, and relationships in the document. Let $\mathcal{A} = \{A_1, A_2, \ \ldots, \ A_n\}$ be a set of corresponding features from each data sample. The k-th value of a feature p is denoted by A_p^k. Features consist of topics, metadata, and mid-level structural units (entities, events, relationships etc.) of a data sample that can infer further higher order structures from them in a bottom-up manner. The goal of using topics and structural units as features is to infer an explainable understanding of how different data samples are similar (or, dissimilar). A data sample $\mathbf{x}_{\mathrm{j}}^{\mathrm{i}}$ is a combination of any subset of $\mathcal{A}$. Features are generated automatically in two steps - 1) a textual description of each data sample is generated from any modality; 2) topics, entities, and events are extracted from the textual descriptions and are considered as weak labels for two reasons. *First,* the quality of the extracted structural units rely on the choice of the extraction models, and can be noisy. *Second,* output generated from the modality specific textual descriptors can be ambiguous and noisy.

For our approach, first, we utilize the existing neural network approaches to find a translation from different modalities of data to a textual representation. Then, we create a data information network by connecting data samples to their features via their interactions. Finally, we construct a structure-infused textual representation, by jointly embedding in a single space the data samples, the features in which these data samples are similar, and the

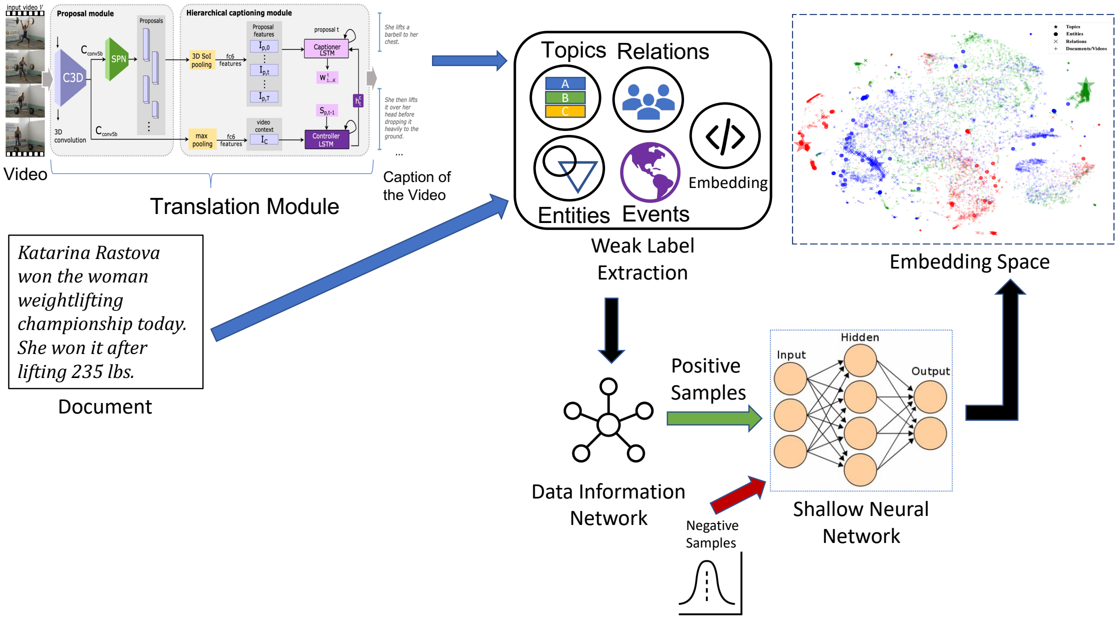


Figure 5.1. Architecture for Weakly Supervised Joint Embedding for Multi-modal Information Retrieval. Translation module is a two-level dense video captioning model from [131].

similarity labels associated with them. We define a multi-task learning objective capturing the interaction information, by aligning the representation of the data samples, defined by their textual content, with the representation of structural features, based on their on their common relations. Moreover, we formalize novelties or data shift that occurs during test time for retrieval task. We also characterize novelties and include appropriate response for different types of novelties.

5.3.2 Translation Module

Videos → Textual Description. To extract an initial representation of videos, we resort to dense video captioning (DVC). We will use a version of dense video captioning described in [131]. DVC localizes distinct events in video streams and generates a description for that. As a feature extraction stage, it uses 3D convolutional network (C3D) to encode all incoming frames. For identifying the event boundaries, maintaining the temporal information

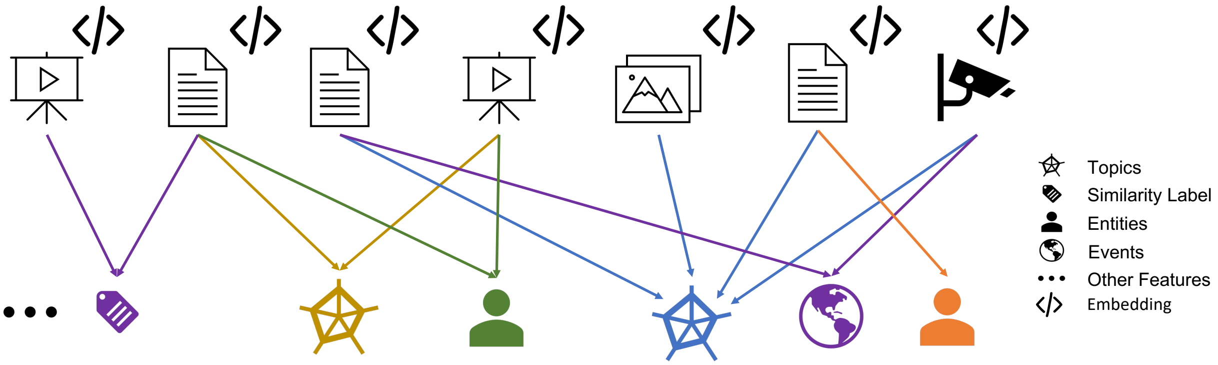


Figure 5.2. Data Information Network

is important and [131] preserves this using convolution and pooling in spatiotemporal space. Using the features from the first stage, in the proposal network, variable-length temporal event proposals are generated and in the final captioning module, they generate a caption for those proposals.

After we have a caption for the whole video, we create the information network from the textual descriptions of all the data samples.

5.3.3 Information Network across Data Samples

There are multiple direct relationships among the data samples and their features. Features can have semantic relationships between them. We define a simple data information graph $G = \{V, E\}$ consisting of several different types of vertices and edges, as follows -

- Let $A_T \subset V$ denote the set of the *topics*.
- Let $A_n \subset V$ denote the set of the *named entities*. A_n is derived from a knowledge base such as, NELL [132], YAGO [133], Wikidata.
- Let $A_{event} \subset V$ denote the set of the *events*. A_{event} is a sentence describing certain real world events.
- Let $x \subset V$ denote the set of the *data samples*. x has a *data modality* attribute.

- Let $A_{sim} \subset V$ denote the set of *user defined similarity labels*.

The graph vertices are connected via a set of edges described hierarchically, as follows:

- $E_{xA_T} \subset E$: All data samples are connected to their corresponding topics. Note that a data sample can be connected to more than one topic.

- $E_{xA_{event}} \subset E$: All data samples are connected to the events that it describes.

- $E_{xA_n} \subset E$: All data samples are connected to the entities it describes. Note that an entity may be described by many different data samples.

- $E_{xA_{sim}} \subset E$: If user has defined a similarity between two data samples, both of them are connected to that similarity label.

In addition to the relations expressed in the graph, all the graph nodes are also associated with textual content. Let A_{text} denote the set of the *textual representations* of the data samples. Topics, entities, similarity label, and events also have their own text representation.

5.3.4 Multi-Task Learning

After we have defined the information graph, we need to design the embedding function to map the graph objects into a low dimensional vector space, such that the graph relationships are preserved in the embedding space. In the embedding space, the relations originally defined over the vertices in the information graph are expressed as a similarity score between the vectors representing these vertices. The relationships between the features themselves are also expressed as a similarity score between the vectors representing them. To force these relationship constraints on the data samples, we consider this as a multi-task learning problem, over all the relations in the graph. Jointly learning over all the relationships allows the weak labels to propagate through elements and enforce multiple degrees of similarities. For example, if a document and a video, or two documents have same topic, they should have similar embedding. In parallel, if the two data samples are discussing about the same event, they should have similar embedding. Our embedding function should jointly reflect these similarities.

For each individual graph relation, R, we can define the learning objective as follows:

$$L_R = \sum_{\mathrm{i}} L(o_{\mathrm{i}}, s^p_{\mathrm{i}}, s^n_{\mathrm{i}}) \tag{5.1}$$

$$L(o_{\mathrm{i}}, s^p_{\mathrm{i}}, s^n_{\mathrm{i}}) = \; y \log sim(o_{\mathrm{i}}, s^p_{\mathrm{i}})) + \;\; (1-y)\log(1 - sim(o_{\mathrm{i}}, s^n_{\mathrm{i}}))) \tag{5.2}$$

where $sim(o_{\mathrm{i}}, s^p_{\mathrm{i}}) = \sigma(\mathrm{e}_{o_{\mathrm{i}}} \cdot \mathrm{e}_{s^p_{\mathrm{i}}})$; $sim(o_{\mathrm{i}}, s^n_{\mathrm{i}}) = \sigma(\mathrm{e}_{o_{\mathrm{i}}} \cdot \mathrm{e}_{s^n_{\mathrm{i}}})$

In equation 5.1, for each object, o_{i} in the graph participating in relation R, s^p_{i} and s^n_{i} refers to positive and negative examples, respectively. $\mathrm{e}_{o_{\mathrm{i}}}$ refers to the vector embedding of the graph object o_{i}, and y is the label. The objective of the model is to maximize the similarity with a positive example and minimize the similarity with a negative example. So, $y = 1$ for $(o_{\mathrm{i}}, s^p_{\mathrm{i}})$ pairs and $y = 0$ for $(o_{\mathrm{i}}, s^n_{\mathrm{i}})$ pairs since they have been sampled from the noise distribution.

Next, we introduce different learning objectives associated with different relations.

Features to Features **($A_T A_T / A_n A_n / A_{event} A_{event}$):** These objective functions place the same type of features with similar context together in the embedding space. The similarity in context refers to similar word or sentence embedding. If the topics, named entities, or events have embedding value within a certain threshold, they are considered similar.

Data Sample to Data Sample **($x^D x^V / x^D x^D / x^V x^V$):** Currently, in this work, data samples refer to videos (x^V) and documents (x^D). This objective function maximizes the similarity of the data samples pair (x_{i}, x_{j}) with the motivation that data objects discussing about *similar events between similar entities on similar topics* should be semantically similar. We select the positive pairs for respective objective function in following ways:

1. xx^{topics}. If data samples are annotated and *topic* annotations are available, we pair the data samples which belong to the same group.

2. $xx^{events-entities}$. For named entities and events, we can consider them together to select the pairs since entities separately does not contribute towards two document or videos being similar. So, data samples discussing about *common events* between a threshold number of *common entities* belong to the same pair.

3. xx^{label}. Two data samples with a user-provided positive similarity between them should have similar embedding.

4. $xx^{embedding}$. If initially two data samples have text embedding representation ($a_{text} \subset A_{text}$) within a certain threshold, they are placed in the same embedding space. The threshold is determined empirically for each application domain.

Data Samples to Features ($xA_T/xA_{\text{event}}/xA_n$): This objective tends to maximize the similarity of data samples to their features. For example, if we only consider topics as the only feature, we want to place the data samples belonging to a certain topic closer to that topic. In the embedding space, the data sample vectors should be closer to the topic embedding vectors.

Joint Objective Function. Finally, we combine the loss functions of all the learning objectives to define our joint embedding loss function. The set of possible learning objectives, $O = \{A_TA_T,\ A_nA_n,\ A_{\text{event}}A_{\text{event}},\ x^Dx^V,\ x^Dx^D,\ x^Vx^V,\ xA_T,\ xA_{\text{event}},\ xA_n\}$ is expandable as we consider more features in future. We experimented with different combinations of these objective functions. So, the combined loss function is

$$L_{total} = \sum_{\text{i} \in O_s, O_s \subset O} \lambda_{\text{i}} L_{\text{i}} \tag{5.3}$$

Here, O_s refers to the selected objective functions and λ_{i} refers to the weight applied to the objective function i. For our experiments we set the value of λ_{i} to 1 for all the objectives.

Initial Representation of Graph Elements: For all objects in the graph, the initial representation is chosen from different representations for text. We experimented with different initial representations for text, and then use a hidden layer to map the initial representations in the joint embedding space. This linear layer filters out the important features from the initial representation for the joint embedding. For an initial representation, t of a text, the hidden layer computes its embedding e as follow.

$$\text{e} = f(Wt + b) \tag{5.4}$$

5.3.5 Reasoning Over the Data Information Network

Our end goal is to use the vector representations of the data samples and features to extract all relevant data samples from the database given a particular data sample. The relevance can be defined directly over the embedding space, by comparing the similarity of the vectors representing respective data samples. We can calculate a relevance score by taking the graph structure into account, by exploiting inter-dependencies among features.

Weak Supervised Baseline.

To use the information graph that we built, we use the information from graph directly without any learning. This is achieved by counting the paths from one data sample to a given data sample or a given feature. Let $P(a, b)$ define the set of paths from given data sample a to another data sample b. Each path is associated with a weight w. Weights are assigned to each path considering the features that exists in the path. Initially, we can consider all weights to 1. But in reality some degrees of similarity have higher precedence. For example, a user defined similarity should have the highest priority. We hypothesize the following feature order to assign the weights based on the priority assigned by domain experts -

$$A_{sim} > A_{text} > A_T > A_{\text{event}}, A_n \mid A_u \tag{5.5}$$

So a path with $E_{xA_{sim}}$ has a higher weight than a path with $E_{xA_{text}}$. Given the graph G, edges connect a data sample to its features, and then features to other data samples having the same features. The relevance score between a and b is then defined as:

$$Rel(a, b) = \frac{\sum_{\text{i} \in P(a,b)} w_{\text{i}}}{\sum_{b \in B} \sum_{\text{i} \in P(a,b)} w_{\text{i}}} \tag{5.6}$$

where B is the set of all the data samples in the database.

In case we need to find all the data samples given a feature, we can retrieve them directly from the graph structure. Let $N_p(f, b)$ define the number of paths from given feature f to a data sample b. The relevance score between f and b is then defined as:

$$Rel(f, b) = I * N_p(f, b) \tag{5.7}$$

where I is an indicator variable. I = 0, if there is no path between f and b, otherwise I = 1.

Similarity Based Score.

Given a data sample, or a feature a and their embedding e_a the relevance score with other data sample b with embedding e_b is:

$$Rel(a, b) = sim(\mathrm{e}_a, \mathrm{e}_b) \tag{5.8}$$

where $sim()$ is the cosine distance between the vectors representing the given data sample, or feature and the other data sample.

5.4 Experiments

The first set of experiment compares the embedding approach for single modality information retrieval (text → text) when there is only topics are available as feature. We call this model **D**ata with **T**opics to **D**ata **Vec**tors **(DT2DVec)**. We experimented with the following representations for caption document of videos, documents, and topics -

- random initialization of the document,
- average of pre-trained GloVe word embedding (300d) of filtered tokens from the document,
- Skip-Thought [134] for capturing the global context of the document,
- BERT-Base uncased model for generation of text representation T from the token sequence t of the document.

These initial representations are mapped into a hidden layer to map them into the joint embedding space as part of the retrieval model, as shown in equation 5.4.

5.4.1 Negative Sampling

As in [135], we used negative sampling to train the model. Our goal is to minimize the similarity of the target object, o_i and samples drawn from the noise distribution, $\mathcal{P}_n(o_\mathrm{i})$ with k negative samples for each data sample. DT2DVec investigated with a number of choices for $\mathcal{P}_n(o_\mathrm{i})$.

1. Following [135], we pick $\mathcal{P}_n(o_\mathrm{i})$ from the uniform distribution of the objects in the dataset. Objects in the dataset consist of the documents from video captions, text, and topics of the texts and videos. The uniform distribution of the objects in the dataset d is $U(d)$ raised to the 3/4rd power with $U(d)$ being the frequency of objects in the respective dataset. Documents are different from words, as words can appear multiple times in a document where often documents do not appear multiple times in a dataset.

2. Given, we have the annotated topics for documents, we consider the noise distribution of each topic t, $\mathcal{P}_n(t)$ from the document samples of other topics. Any data sample that is not annotated with topic t belongs to $\mathcal{P}_n(t)$.

For batch training, DT2DVec adopted the following approaches: **(2a)** Let us assume there are p number of positive pairs in a batch, and the set of topics of these pairs is POS_T. If the mode of POS_T is topic t, then we pick negative examples from $\mathcal{P}_n(t)$ for this batch. The intuition behind the approach is the closer graph objects in the embedding space should have similar distribution; **(2b)** We select variable number of negative examples for each batch. Negative examples are selected from $\mathcal{P}_n(t)$ of each topic in the positive pairs, weighted by the number of positive pairs from each topic.

Table 5.1. Performance Comparison Results of DT2DVec

	LSA	LDA	DT2DVec
Inter-similarity	0.76	0.66	0.61
Intra-similarity	0.45	0.28	0.047

5.4.2 Dataset and Experimental Setup

For the performance evaluation of DT2DVec, we used the 20 Newsgroups dataset [136] with twenty annotated topics. We compared DT2DVec with two baseline topic modeling approaches - LSA [137] and LDA [138]. For evaluation, we split each document into two parts, and test if **(1)** the topics of the first half are similar to topics of the second half (inter-similarity); **(2)** halves of different documents are mostly dissimilar (intra-similarity). We use cosine similarity to measure the difference between the two vectors of half document topics. For inter-similarity, higher similarity score is better. For intra-similarity, the lower the similarity, the better the vectors are. We present the result for the random initialization of initial representation of text in Table 5.1. We used Pytorch [139] with binary cross entropy to train the unsupervised retrieval model. Mini-batch gradient descend was used for optimization with SGD [140].

5.4.3 Results.

(DT2DVec - Rand) performed significantly better in recognizing the dissimilar documents than the baseline models, LSA and LDA. There was around 43% improvement in similarity score for dissimilar documents whereas LSA outperformed our method by 10% for simiar documents. Figure 5.3 shows the embedding space of the documents using the DT2DVec model.

5.5 Conclusion and Future Works

This paper proposed a *weakly supervised open world learning* framework for multi-modal information retrieval. Our methods involve no human annotation, show promising performance compared to unsupervised approaches, and formalize novelties encountered during testing. In the future, we would test our novelty characterization, detection and adaptation framework with different datasets. We would also include different modalities in our framework, and would test the capability of the framework for domain generalization.

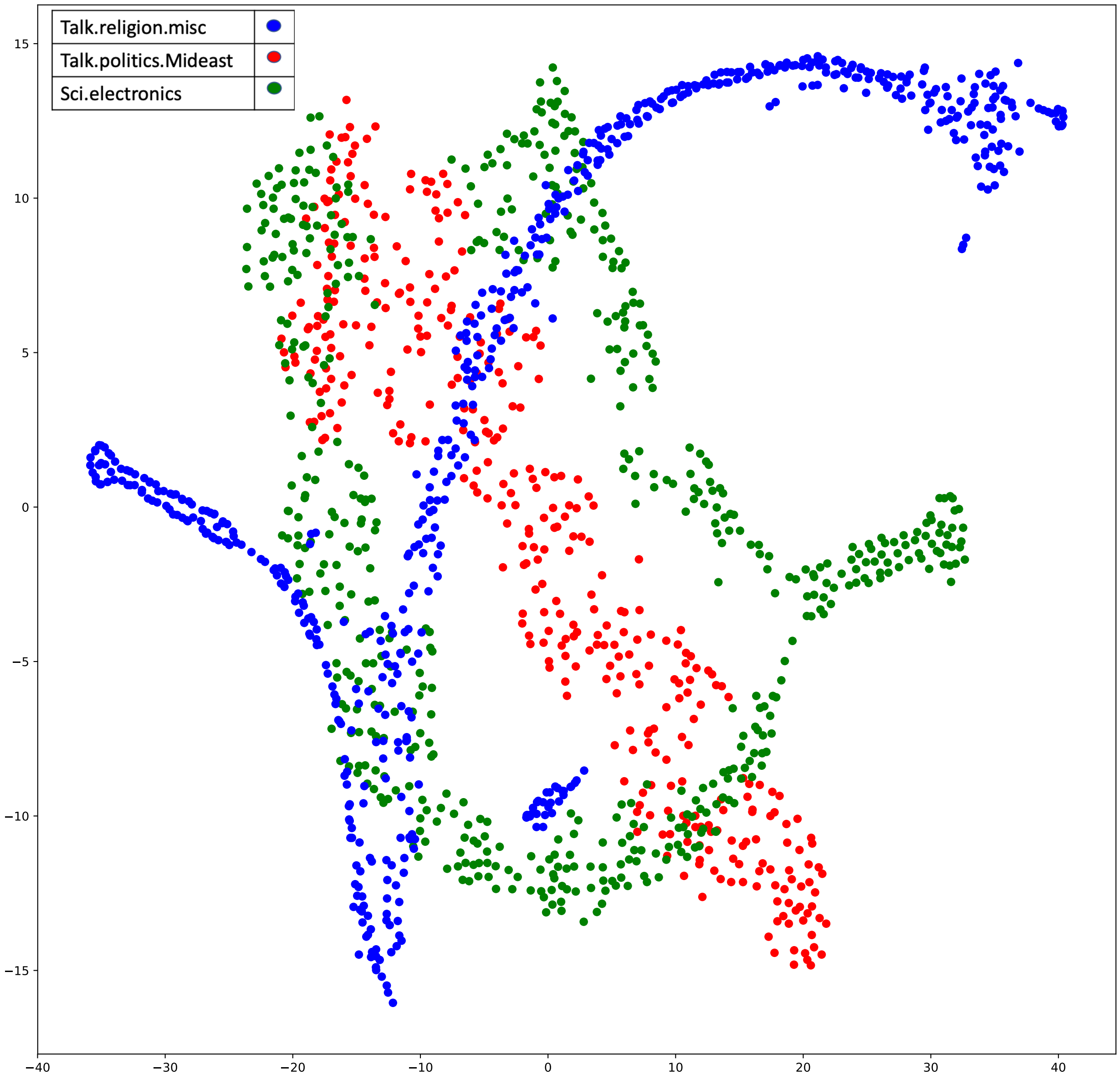


Figure 5.3. t-SNE Embedding of Documents using DT2DVec with Random Initial Representation of Text

6. UNCERTAINTY MANAGEMENT IN DATA-DRIVEN APPLICATIONS

6.1 Formalization of Novelties in Multimodal Information Retrieval

In this work, we discuss and formalize novelties in multimodal retrieval task in terms of data shift. As part of our framework, we add a novelty detection and characterization criterion. Finally, we design a pre-training strategy for handling out-of-distribution inputs. It has three parts in our setting. We pretrain the video encoder separately on video captioning task. The weak label generation does not need any pre-training. Then the graph object representations will be pre-trained under the relationship objectives in the final stage.

6.1.1 Task Definition

Existing works in multi-modal information retrieval defines it in several different ways. In *supervised setting*, following our previous notations, Let the training data be $\mathcal{D}_{tr} = \{(\mathbf{x}^{\mathrm{i}}_{\mathrm{j}}, y^{\mathrm{i}}_{\mathrm{j}})\}^{n_{\mathrm{i}}}_{\mathrm{j}=1}$, where n_{i} is the number of samples in i-th modality, $\mathbf{x}_{\mathrm{j}} \in X$ is a training example following the training distribution $P_{tr}(\mathbf{x})$. $y_{\mathrm{j}} \in Y_{tr}$ is the corresponding class label of $\mathbf{x}_{\mathrm{j}}$ and Y_{tr} is the set of all class labels that appear in $\mathcal{D}_{tr}$. Each modality have their own training distribution $P_{tr}(\mathbf{x}^{\mathrm{i}})$, but for simplicity purpose, we are going to denote training distribution as only $P_{tr}(\mathbf{x})$. For *unsupervised setting*, y_{j} is absent in $\mathcal{D}_{tr}$. In our *weakly supervised setting*, class labels are still absent, but the extracted features act as weak labels and amplifies similarity signal among data samples through the network structure. The retrieval task refers to estimating probability of a data sample being relevant to a query given the data sample and a query sample, $P(R \mid x_p, x_q)$, where $x_p, x_q \in X$, and R is the corresponding relevance label.

6.1.2 Data Shift in Multimodal Data Retrieval Task

Following the discussion in [129] and [141], we define the three main types of data shift that can happen during testing for Multimodal Data Retrieval Task.

Covariate shift refers to the distribution change of the input variable x between training and test phases, i.e., $P_{tr}(R \mid x_p, x_q) = P_{\text{te}}(R \mid x_p, x_q)$ and $P_{tr}(\mathbf{x}) \neq P_{te}(\mathbf{x})$. This can refer to change in application domain while still dealing with the same modalities in P_{tr}. This also can occur if user starts to phrase their queries differently.

Prior probability shift refers to the distribution change of the class variable y, or the relevance variable R, or the weak feature variables $\mathcal{A}$, i.e., in our framework, $P_{tr}(x_p \mid R, x_q) = P_{\text{te}}(x_p \mid R, x_q)$ and $(P_{tr}(R) \neq P_{\text{te}}(R)$ or $P_{tr}(A) \neq P_{\text{te}}(A))$. In WesJem, this includes not having extracted weak features from a data sample during testing.

Concept drift refers to the change in the posterior probability distribution between training and test phases, i.e., $P_{tr}(R \mid x_p, x_q) \neq P_{\text{te}}(R \mid x_p, x_q)$ and $P_{tr}(\mathbf{x}) = P_{\text{te}}(\mathbf{x})$. This can be a temporal effect or user requirement has changed over time.

Besides the three types of data shifts, multimodal retrieval faces one other type of change during testing, i.e., data samples that do not belong to the modalities that the framework can handle. These are *novelty* or *novel instances.* This is closely related to covariate shift and some framework may handle novel instances as part of a known class.

6.1.3 Novelty Detection in WeSJem

We use the *data information network* to detect the changes between pre-novelty and post-novelty environments. During inference with a novelty introduction, after the translation and weak feature extraction, we have the post-novelty graph. We can use existing node discovery techniques to detect change from the training time information network. In case of a *novel modality*, our proposed framework would either identify the new weak features (different from training time), or tackle the new modality as part of the training distribution. In the later case, we may see a decline in the model performance.

Definition 6.1.1. ***(Novel Instance).*** *A test instance x is novel if $G(V_{P_{tr}+\mathbf{x}}, E)$ is different from $G(V_{P_{tr}}, E)$. This can be explained as having a knowledge base for the weak features during training time ($\mathcal{A}_{tr}$). If during inference, we discover weak features that are absent in $\mathcal{A}_{tr}$, we consider the instance as novel.*

6.1.4 Novelty Characterization for Multimodal Information Retrieval

Definition 6.1.2. ***(Characterization of Novelty).*** *Characterization of Novelty is the description of the novelty, according to which appropriate course of actions are taken to respond to the novelty. We characterize novelty based on the data shift variations -*

1. *Covariate shift with change in application domain with the modalities for which translation module is available (covar-1).*
2. *Prior probability shift with novel weak features (prior-1).*
3. *Prior probability shift with no weak features (prior-2).*
4. *Prior probability shift with novel relevance label (prior-3).*
5. *Temporal concept drift with previously relevant data being non-relevant (concept-1).*
6. *Covariate shift with new modality introduction (covar-2).*

6.1.5 Novelty Response in WeSJem

For a generalized response, we propose to build a pre-trained retrieval model from WeSJem to deal with the out-of-distribution (OOD) inputs. We adopt a three level training strategy for our model. For the first stage, we pre-train the translation module (DVC) with video captioning and video retrieval task, following the strategy in [123] and [142]. JEDDiNet is trained on both the ActivityNet Captions [143] dataset and MSR-VTT [142] dataset. MSR-VTT has open domain video clips, and each clip has 20 captioning sentences labeled by human. For both cases, we used the SPN module from [142] trained with the temporal annotation of ground truth segments in the ActivityNet Captions dataset with Sports-1M pretrained C3D weight initialized in [144]. For the text encoder, we choose between the pre-trained BERT [126] Base uncased model and the pre-trained skip-thought model [134]. Next, we extract the weak features from the dense captions and document text using topic and event extraction models such as [145, 146]. Finally, we train our weakly supervised model using the joint objective loss function.

During inference, the translation module is able to generate captions for OOD inputs, which includes input from new modalities as in *(covar-2)*, i.e., image or LIDAR. Both image and LIDAR modality can be handled as a variant of the video translation module. Both BERT and skip-thought can generate text embedding for any textual input. This takes care of the novel weak features *(prior-1)*. Finally, the linear layers in WeSJem would be able to map the OOD inputs into the pre-trained joint embedding space. The final similarity score between data samples in the embedding space would produce the relevance between new samples.

With the encounter of a *(prior-2)* novelty, if the system is allowed to $learn$, in proposed joint embedding model, the set of selected learning objectives becomes,

$$O_s = \{x^D x^V, x^D x^D, x^V x^V\}$$

where only the $\mathbf{xx^{embedding}}$ objective function is considered, as each data sample would include an initial representation from the textual descriptions.

Finally, when encountered with any of the last three types of novelties, an information retrieval system needs to re-learn. In case of a novel modality introduction, the long-time response is to $learn$ or $gather$ a new translation method. In our model, we include this as a **Relevance Feedback** module. The new relevance label provided by a human annotator holds more importance than previous relevance labels. To make a distinction between this newly provided label and old label between (x_p, x_q), during re-training, we encode this by assigning more priority to the new similarity label, $A^{new}_{sim} > A^{old}_{sim}$.

7. CONCLUSION

This dissertation provides comprehensive study of strategies to enhance data-driven decision making and multimodal information retrieval in open-world dynamic environments. The research addresses various challenges related to heterogeneous and missing data sources, scalability, data integration, relevance learning, and uncertainty management, aiming to empower decision-makers in extracting valuable insights from diverse multimodal data sources.

One of the key contributions is the introduction of the SKOD framework, which enables the continuous building of a multimodal relational knowledge base and facilitates the delivery of decision-making information from various data sources. SKOD effectively addresses scalability and data completeness challenges by utilizing streaming brokers and RDBMS capabilities. It leverages semantic features as a powerful content representation and ensures continuous delivery of data while adapting to user interests and missing modalities over time. For data integration, it relied on schema mapping and mediator based SQL-JOIN for a scalable data delivery and exact matching with EARS. Additionally, a novel human attribute recognition model is developed to extract fine-grained properties from unstructured text, specifically addressing human attribute extraction. This model outperforms standalone language models in detecting attributes, thereby contributing to more accurate and intelligent text processing for human consumption. Real-world system prototypes are built to assist law enforcement officers in automating investigations and finding missing persons, demonstrating the effectiveness of the proposed framework.

To handle data integration challenges in open-world applications, we introduced the FemmIR approach, using deep neural network which employs feature-centric multimodal information retrieval with graph matching and delivers a ranked list of relevant data to user information need. We also proposed a weakly supervised representation learning approach, WeSJem, for learning an embedding function from features of disconnected sources, eliminating the need for annotations. FemmIR performed comparatively with EARS while using the historical queries and achieving an approximate matching.

Our final contribution for decision making information extraction from multimodal data was to charaterize uncertainties in a dynamic environment based on data drift. Novelty detection and adaptation techniques are proposed within the WeSJem framework. Furthermore, we proposed methodologies for measuring novelty difficulty in planning domains [147] and a novel intrinsic complexity metric in distributed perception domains [148]. These metrics provide insights into the adaptability and complexity of learning-based decision making in dynamic environments.

Overall, the contributions presented in this dissertation provide valuable advancements in extracting decision-making information from diverse multimodal data sources in open-world environments. The proposed frameworks, models, prototypes, and metrics offer practical solutions to the challenges faced in real-world applications, ultimately enhancing decision-making processes and adaptability in dynamic and uncertain scenarios.

7.1 Future Works

Based on the conclusions drawn from the current research, my future goals are centered around addressing the lingering research questions in data-driven decision making: (1) How to model users information need in a robust and efficient manner? (2) How to avoid bias, increase trust and privacy in recommended results?

To complete the life-cycle of situational knowledge delivery, we still have the following challenges in modeling users information need: (1) user requirement is not always obvious or explicitly stated, (2) user can be interested in multiple types of events and knowledge bases with varying probabilities, (3) learning algorithms need to adapt to changing user preferences with time. I aim to develop novel algorithms using techniques such as active learning and reinforcement learning that can accurately capture and predict users preferences based on their behavior, interactions, and feedback. I aim to jointly model user preferences with the source content and context to deliver more accurate situational knowledge. Understanding the features that drive user preferences, and leveraging this knowledge to improve personalized recommendations and user experience, has applications in education, student advising, classroom teaching, e-commerce, healthcare, etc.

With the rise in the volume of multimodal data being generated and consumed, there is an emerging demand among users to comprehend the factors underlying recommendations, as well as the significance and reliability of the information they are accessing. This is especially important in sensitive domains such as healthcare, finance, and legal decision-making to allow for tracking, cross-checking with social contexts, and verification. To address that, I aim to connect our proposed data integration and adaptability models with explainable models and trustworthy AI.

My long-term goal is to create intelligent systems that can reason, learn and cooperate with humans to improve the standard of living by utilizing the vast amounts of data available in the modern era. My focus is to devise new algorithms and methods that can make a significant impact on society, leverage existing scientific advancements, and address real-world challenges.

REFERENCES

[1] K. Solaiman, T. Sun, A. Nesen, B. Bhargava, and M. Stonebraker, “Applying machine learning and data fusion to the missing person problem,” *Computer*, vol. 55, no. 06, pp. 40–55, Jun. 2022, ISSN: 1558-0814. DOI: 10.1109/MC.2022.3145507.

[2] J. Redmon and A. Farhadi, “Yolov3: An incremental improvement,” *arXiv preprint arXiv:1804.02767*, 2018.

[3] C. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. Bethard, and D. McClosky, “The Stanford CoreNLP natural language processing toolkit,” in *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, Baltimore, Maryland: Association for Computational Linguistics, Jun. 2014, pp. 55–60. DOI: 10.3115/v1/P14-5010. [Online]. Available: https://www.aclweb.org/anthology/P14-5010.

[4] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent dirichlet allocation,” *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003, ISSN: 1532-4435. [Online]. Available: http://dl.acm.org/citation.cfm?id=944919.944937.

[5] S. Palacios, V. Santos, E. Barsallo, and B. Bhargava, “Miostream: A peer-to-peer distributed live media streaming on the edge,” *Multimedia Tools and Applications*, Jan. 2019, ISSN: 1573-7721. DOI: 10.1007/s11042-018-6940-2. [Online]. Available: https://doi.org/10.1007/s11042-018-6940-2.

[6] S. Poria, E. Cambria, N. Howard, G.-B. Huang, and A. Hussain, “Fusing audio, visual and textual clues for sentiment analysis from multimodal content,” *Neurocomput.*, vol. 174, no. PA, pp. 50–59, Jan. 2016, ISSN: 0925-2312. DOI: 10.1016/j.neucom.2015.01.095. [Online]. Available: http://dx.doi.org/10.1016/j.neucom.2015.01.095.

[7] G. L. Foresti, M. Farinosi, and M. Vernier, “Situational awareness in smart environments: Socio-mobile and sensor data fusion for emergency response to disasters,” *J. Ambient Intelligence and Humanized Computing*, vol. 6, no. 2, pp. 239–257, 2015.

[8] G. Meditskos, S. Vrochidis, and I. Kompatsiaris, “Description logics and rules for multimodal situational awareness in healthcare,” in *MMM (1)*, ser. Lecture Notes in Computer Science, vol. 10132, Springer, 2017, pp. 714–725.

[9] O. Adjali, M. D. Hina, S. Dourlens, and A. Ramdane-Cherif, "Multimodal fusion, fission and virtual reality simulation for an ambient robotic intelligence," in *ANT/SEIT*, ser. Procedia Computer Science, vol. 52, Elsevier, 2015, pp. 218–225.

[10] Y. Zhu, J. J. Lim, and L. Fei-Fei, "Knowledge acquisition for visual question answering via iterative querying," in *CVPR*, IEEE Computer Society, 2017, pp. 6146–6155.

[11] Q. Wu, P. Wang, C. Shen, A. R. Dick, and A. van den Hengel, "Ask me anything: Free-form visual question answering based on knowledge from external sources," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2016, pp. 4622–4630. DOI: 10.1109/CVPR.2016.500. [Online]. Available: https://doi.org/10.1109/CVPR.2016.500.

[12] D. Kang, P. Bailis, and M. Zaharia, "Blazeit: Optimizing declarative aggregation and limit queries for neural network-based video analytics," *PVLDB*, vol. 13, no. 4, pp. 533–546, 2019. [Online]. Available: http://www.vldb.org/pvldb/vol13/p533-kang.pdf.

[13] D. B. Nguyen, A. Abujabal, N. K. Tran, M. Theobald, and G. Weikum, "Query-driven on-the-fly knowledge base construction," *Proc. VLDB Endow.*, vol. 11, no. 1, pp. 66–79, Sep. 2017, ISSN: 2150-8097. DOI: 10.14778/3151113.3151119. [Online]. Available: https://doi.org/10.14778/3151113.3151119.

[14] M. Bienvenu, C. Bourgaux, and F. Goasdoué, "Query-driven repairing of inconsistent dl-lite knowledge bases," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, 2016, pp. 957–964. [Online]. Available: http://www.ijcai.org/Abstract/16/140.

[15] X. Dong *et al.*, "Knowledge vault: A web-scale approach to probabilistic knowledge fusion," in *KDD*, ACM, 2014, pp. 601–610.

[16] Y. Chen and D. Z. Wang, "Knowledge expansion over probabilistic knowledge bases," in *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '14, Snowbird, Utah, USA: ACM, 2014, pp. 649–660, ISBN: 978-1-4503-2376-5. DOI: 10.1145/2588555.2610516. [Online]. Available: http://doi.acm.org/10.1145/2588555.2610516.

[17] M. E. Rodrguez, S. Goldberg, and D. Z. Wang, "Sigmakb: Multiple probabilistic knowledge base fusion," *PVLDB*, vol. 9, no. 13, pp. 1577–1580, 2016.

[18] M. L. Itria, A. Daidone, and A. Ceccarelli, "A complex event processing approach for crisis-management systems," *CoRR*, vol. abs/1404.7551, 2014.

[19] G. Cugola and A. Margara, "Processing flows of information: From data stream to complex event processing," *ACM Comput. Surv.*, vol. 44, no. 3, 15:1–15:62, 2012. DOI: 10.1145/2187671.2187677. [Online]. Available: https://doi.org/10.1145/2187671.2187677.

[20] R. Krishna *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *CoRR*, vol. abs/1602.07332, 2016. arXiv: 1602.07332. [Online]. Available: http://arxiv.org/abs/1602.07332.

[21] T.-Y. Lin *et al.*, "Microsoft coco: Common objects in context," in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., Cham: Springer International Publishing, 2014, pp. 740–755, ISBN: 978-3-319-10602-1.

[22] S. Abu-El-Haija *et al.*, "Youtube-8m: A large-scale video classification benchmark," *CoRR*, vol. abs/1609.08675, 2016. arXiv: 1609.08675. [Online]. Available: http://arxiv.org/abs/1609.08675.

[23] Chiao-Fe Shu *et al.*, "Ibm smart surveillance system (s3): A open and extensible framework for event based surveillance," in *IEEE Conference on Advanced Video and Signal Based Surveillance, 2005.*, 2005, pp. 318–323.

[24] B. Tian *et al.*, "Hierarchical and networked vehicle surveillance in its: A survey," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 16, pp. 557–580, Apr. 2015. DOI: 10.1109/TITS.2014.2340701.

[25] D. Kang, J. Emmons, F. Abuzaid, P. Bailis, and M. Zaharia, "Noscope: Optimizing deep cnn-based queries over video streams at scale," *PVLDB*, vol. 10, no. 11, pp. 1586–1597, 2017. DOI: 10.14778/3137628.3137664. [Online]. Available: http://www.vldb.org/pvldb/vol10/p1586-kang.pdf.

[26] M. R. Anderson, M. J. Cafarella, G. Ros, and T. F. Wenisch, "Physical representation-based predicate optimization for a visual analytics database," *CoRR*, 2018. arXiv: 1806.04226. [Online]. Available: http://arxiv.org/abs/1806.04226.

[27] K. Hsieh *et al.*, "Focus: Querying large video datasets with low latency and low cost," in *13th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2018, Carlsbad, CA, USA, October 8-10, 2018*, A. C. Arpaci-Dusseau and G. Voelker, Eds., USENIX Association, 2018, pp. 269–286. [Online]. Available: https://www.usenix.org/conference/osdi18/presentation/hsieh.

[28] I. Xarchakos and N. Koudas, "SVQ: streaming video queries," in *Proceedings of the 2019 International Conference on Management of Data, SIGMOD Conference 2019, Amsterdam, The Netherlands, June 30 - July 5, 2019*, P. A. Boncz, S. Manegold, A. Ailamaki, A. Deshpande, and T. Kraska, Eds., ACM, 2019, pp. 2013–2016. DOI: 10.1145/3299869.3320230. [Online]. Available: https://doi.org/10.1145/3299869.3320230.

[29] Y. Lu, A. Chowdhery, and S. Kandula, "Optasia: A relational platform for efficient large-scale video analytics," in *Proceedings of the Seventh ACM Symposium on Cloud Computing, Santa Clara, CA, USA, October 5-7, 2016*, M. K. Aguilera, B. Cooper, and Y. Diao, Eds., ACM, 2016, pp. 57–70. DOI: 10.1145/2987550.2987564. [Online]. Available: https://doi.org/10.1145/2987550.2987564.

[30] H. Zhang, G. Ananthanarayanan, P. Bodk, M. Philipose, P. Bahl, and M. J. Freedman, "Live video analytics at scale with approximation and delay-tolerance," in *14th USENIX Symposium on Networked Systems Design and Implementation, NSDI 2017, Boston, MA, USA, March 27-29, 2017*, A. Akella and J. Howell, Eds., USENIX Association, 2017, pp. 377–392. [Online]. Available: https://www.usenix.org/conference/nsdi17/technical-sessions/presentation/zhang.

[31] Y. Zhang and A. Kumar, "Panorama: A data system for unbounded vocabulary querying over video," *Proceedings of the VLDB Endowment*, vol. 13, no. 4, pp. 477–491, 2019.

[32] C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 3, pp. 453–465, 2013.

[33] B. M. Lake, R. R. Salakhutdinov, and J. Tenenbaum, "One-shot learning by inverting a compositional causal process," in *Advances in neural information processing systems*, 2013, pp. 2526–2534.

[34] X. Liu *et al.*, "Hydraplus-net: Attentive deep features for pedestrian analysis," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1–9.

[35] M. Stonebraker *et al.*, "Surveillance video querying with a human-in-the-loop," in *Proceedings of the Workshop on Human-In-the-Loop Data Analytics with SIGMOD*, 2020.

[36] C. Fellbaum, *WordNet: An Electronic Lexical Database*. Bradford Books, 1998.

[37] T. Mikolov, K. Chen, G. Corrado, and J. Dean, *Efficient estimation of word representations in vector space*, 2013. arXiv: 1301.3781 [cs.CL].

[38] Z. Wu and M. Palmer, "Verb semantics and lexical selection," *arXiv preprint cmp-lg/9406033*, 1994.

[39] W. Yin, J. Hay, and D. Roth, "Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach," *arXiv preprint arXiv:1909.00161*, 2019.

[40] N. s Reimers and I. Gurevych, *Sentence-bert: Sentence embeddings using siamese bert-networks*, 2019. arXiv: 1908.10084 [cs.CL].

[41] E. Loper and S. Bird, "Nltk: The natural language toolkit," in *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1*, ser. ETMTNLP '02, Philadelphia, Pennsylvania: Association for Computational Linguistics, 2002, pp. 63–70. DOI: 10.3115/1118108.1118117. [Online]. Available: https://doi.org/10.3115/1118108.1118117.

[42] A. Williams, N. Nangia, and S. Bowman, "A broad-coverage challenge corpus for sentence understanding through inference," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 1112–1122. DOI: 10.18653/v1/N18-1101. [Online]. Available: https://www.aclweb.org/anthology/N18-1101.

[43] Q. Chen, J. Du, S. Kim, W. J. Wilbur, and Z. Lu, "Combining rich features and deep learning for finding similar sentences in electronic medical records," *Proceedings of the BioCreative/OHNLP Challenge*, pp. 5–8, 2018.

[44] C. L. GOH and Y. LEPAGE, "Finding similar examples for aiding academic writing using sentence embeddings," 2020.

[45] J. Wang, X. Zhu, S. Gong, and W. Li, "Attribute recognition by joint recurrent learning of context and correlation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 531–540.

[46] G. Pearson, M. Gill, S. Antani, L. Neve, and G. Thoma, "People locator: A system for family reunification," *IT Professional*, vol. 14, no. 03, pp. 13–21, May 2012, ISSN: 1941-045X. DOI: 10.1109/MITP.2012.25.

[47] R. S. Ferreira, C. G. de Oliveira, and A. A. Lima, "Myosotis: An information system applied to missing people problem," in *Proceedings of the XIV Brazilian Symposium on Information Systems*, 2018, pp. 1–7.

[48] H.-X. Yu, W.-S. Zheng, A. Wu, X. Guo, S. Gong, and J.-H. Lai, "Unsupervised person re-identification by soft multilabel learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2148–2157.

[49] S. Aggarwal, V. B. RADHAKRISHNAN, and A. Chakraborty, "Text-based person search via attribute-aided matching," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2020, pp. 2617–2625.

[50] Z. Wang, Z. Fang, J. Wang, and Y. Yang, "Vitaa: Visual-textual attributes alignment in person search by natural language," in *European Conference on Computer Vision*, Springer, 2020, pp. 402–420.

[51] M. Khan and A. Jalal, "A fuzzy rule based multimodal framework for face sketch-to-photo retrieval," *Expert Systems with Applications*, vol. 134, May 2019. DOI: 10.1016/j.eswa.2019.05.040.

[52] J. Rupnik and J. Shawe-Taylor, "Multi-view canonical correlation analysis," in *Conference on Data Mining and Data Warehouses (SiKDD 2010)*, 2010, pp. 1–4.

[53] L. Zhang, B. Ma, G. Li, Q. Huang, and Q. Tian, "Generalized semi-supervised and structured subspace learning for cross-modal retrieval," *IEEE Transactions on Multimedia*, vol. 20, no. 1, pp. 128–141, 2017.

[54] Y. Peng, J. Qi, X. Huang, and Y. Yuan, "Ccl: Cross-modal correlation learning with multigrained fusion by hierarchical network," *IEEE Transactions on Multimedia*, vol. 20, no. 2, pp. 405–420, 2017.

[55] W. Wang, R. Arora, K. Livescu, and J. Bilmes, "On deep multi-view representation learning," in *International conference on machine learning*, PMLR, 2015, pp. 1083–1092.

[56] S. Sah, S. Gopalakrishnan, and R. Ptucha, "Aligned attention for common multimodal embeddings," *Journal of Electronic Imaging*, vol. 29, pp. 023 013–023 013, 2020.

[57] K. Li, Y. Zhang, K. Li, Y. Li, and Y. Fu, "Visual semantic reasoning for image-text matching," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4654–4662.

[58] F. Wu *et al.*, "Modality-specific and shared generative adversarial network for cross-modal retrieval," *Pattern Recognit.*, vol. 104, p. 107 335, 2020.

[59] X. Wang, P. Hu, L. Zhen, and D. Peng, "Drsl: Deep relational similarity learning for cross-modal retrieval," *Inf. Sci.*, vol. 546, pp. 298–311, 2021.

[60] S. Palacios *et al.*, "Wip - skod: A framework for situational knowledge on demand," in *Heterogeneous Data Management, Polystores, and Analytics for Healthcare*, Cham: Springer International Publishing, 2019, pp. 154–166, ISBN: 978-3-030-33752-0.

[61] K. Solaiman and B. Bhargava, "Multimodal information retrieval for systems with explicit information needs and object properties (femmir)," [Online]. Available: https://ksolaiman.github.io/files/publications/sigmod-femmir-2023.pdf.

[62] S. Palacios *et al.*, "Wip - skod: A framework for situational knowledge on demand," in *Heterogeneous Data Management, Polystores, and Analytics for Healthcare*, V. Gadepally *et al.*, Eds., Cham: Springer International Publishing, 2019, pp. 154–166, ISBN: 978-3-030-33752-0.

[63] M. Imhof and M. Braschler, "A study of untrained models for multimodal information retrieval," *Information Retrieval Journal*, vol. 21, no. 1, pp. 81–106, 2018.

[64] K. Srinivasan, K. Raman, J. Chen, M. Bendersky, and M. Najork, "Wit: Wikipedia-based image text dataset for multimodal multilingual machine learning," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 2443–2449.

[65] S. Galhotra, A. Fariha, R. Lourenço, J. Freire, A. Meliou, and D. Srivastava, "Dataprism: Exposing disconnect between data and systems," in *Proceedings of the 2022 International Conference on Management of Data*, ser. SIGMOD '22, Philadelphia, PA, USA: Association for Computing Machinery, 2022, pp. 217–231, ISBN: 9781450392495. DOI: 10.1145/3514221.3517864. [Online]. Available: https://doi.org/10.1145/3514221.3517864.

[66] H. Liu, S. Lu, M. Musuvathi, and S. Nath, "What bugs cause production cloud incidents?" In *Proceedings of the Workshop on Hot Topics in Operating Systems*, ser. HotOS '19, Bertinoro, Italy: Association for Computing Machinery, 2019, pp. 155–162, ISBN: 9781450367271. DOI: 10.1145/3317550.3321438. [Online]. Available: https://doi.org/10.1145/3317550.3321438.

[67] E. K. Rezig *et al.*, "Dagger: A data (not code) debugger," in *CIDR*, 2020.

[68] X. L. Dong *et al.*, "From data fusion to knowledge fusion," *Proc. VLDB Endow.*, vol. 7, no. 10, pp. 881–892, Jun. 2014, ISSN: 2150-8097. DOI: 10.14778/2732951.2732962. [Online]. Available: https://doi.org/10.14778/2732951.2732962.

[69] X. Wang, X. L. Dong, and A. Meliou, "Data x-ray: A diagnostic tool for data errors," in *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '15, Melbourne, Victoria, Australia: Association for Computing Machinery, 2015, pp. 1231–1245, ISBN: 9781450327589. DOI: 10.1145/2723372.2750549. [Online]. Available: https://doi.org/10.1145/2723372.2750549.

[70] E. K. Rezig *et al.*, "Dice: Data discovery by example," *Proc. VLDB Endow.*, vol. 14, no. 12, pp. 2819–2822, Jul. 2021, ISSN: 2150-8097. DOI: 10.14778/3476311.3476353. [Online]. Available: https://doi.org/10.14778/3476311.3476353.

[71] R. Angles and C. Gutierrez, "Survey of graph database models," *ACM Computing Surveys (CSUR)*, vol. 40, no. 1, pp. 1–39, 2008.

[72] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[73] S. Cao, W. Lu, and Q. Xu, "Deep neural networks for learning graph representations," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, 2016.

[74] S. Sawant, "Collaborative filtering using weighted bipartite graph projection: A recommendation system for yelp," in *Proceedings of the CS224W: Social and information network analysis conference*, vol. 33, 2013.

[75] S. Gurukar *et al.*, *Multibisage: A web-scale recommendation system using multiple bipartite graphs at pinterest*, 2022. DOI: 10.48550/ARXIV.2205.10666. [Online]. Available: https://arxiv.org/abs/2205.10666.

[76] Y. Wei, X. Wang, L. Nie, X. He, R. Hong, and T.-S. Chua, "Mmgcn: Multi-modal graph convolution network for personalized recommendation of micro-video," in *Proceedings of the 27th ACM International Conference on Multimedia*, ser. MM '19, Nice, France: Association for Computing Machinery, 2019, pp. 1437–1445, ISBN: 9781450368896. DOI: 10.1145/3343031.3351034. [Online]. Available: https://doi.org/10.1145/3343031.3351034.

[77] J. Ren *et al.*, "Matching algorithms: Fundamentals, applications and challenges," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 5, no. 3, pp. 332–350, 2021.

[78] R. Socher, D. Chen, C. D. Manning, and A. Ng, "Reasoning with neural tensor networks for knowledge base completion," *Advances in neural information processing systems*, vol. 26, 2013.

[79] Y. Liang and P. Zhao, "Similarity search in graph databases: A multi-layered indexing approach," in *2017 IEEE 33rd International Conference on Data Engineering (ICDE)*, IEEE, 2017, pp. 783–794.

[80] W. Zheng, L. Zou, X. Lian, D. Wang, and D. Zhao, "Graph similarity search with edit distance constraint in large graph databases," in *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, 2013, pp. 1595–1600.

[81] K. Riesen and H. Bunke, "Iam graph database repository for graph based pattern recognition and machine learning," in *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, Springer, 2008, pp. 287–297.

[82] K. Riesen and H. Bunke, "Approximate graph edit distance computation by means of bipartite graph matching," *Image and Vision computing*, vol. 27, no. 7, pp. 950–959, 2009.

[83] B. Xiao, X. Gao, D. Tao, and X. Li, "Hmm-based graph edit distance for image indexing," *International Journal of Imaging Systems and Technology*, vol. 18, no. 2-3, pp. 209–218, 2008.

[84] Z. Li, J. Tang, L. Zhang, and J. Yang, "Weakly-supervised semantic guided hashing for social image retrieval," *International Journal of Computer Vision*, vol. 128, no. 8, pp. 2265–2278, 2020.

[85] N. C. Mithun, S. Paul, and A. K. Roy-Chowdhury, "Weakly supervised video moment retrieval from text queries," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 592–11 601.

[86] Y. Alaudah, M. Alfarraj, and G. AlRegib, "Structure label prediction using similarity-based retrieval and weakly supervised label mappingstructure label prediction," *Geophysics*, vol. 84, no. 1, pp. V67–V79, 2019.

[87] K. Solaiman and B. Bhargava, "Open-learning framework for multi-modal information retrieval with weakly supervised joint embedding," in *Proceedings of the AAAI Spring Symposium on Designing Artificial Intelligence for Open Worlds, Palo Alto, CA, USA*, Mar. 2022. [Online]. Available: https://usc-isi-i2.github.io/AAAI2022SS/papers/SSS-22_paper_29.pdf.

[88] K. Riesen, M. Neuhaus, and H. Bunke, "Bipartite graph matching for computing the edit distance of graphs," in *International Workshop on Graph-Based Representations in Pattern Recognition*, Springer, 2007, pp. 1–12.

[89] Y. Bai, H. Ding, S. Bian, T. Chen, Y. Sun, and W. Wang, "Graph edit distance computation via graph neural networks," *arXiv preprint arXiv:1808.05689*, 2018.

[90] P. V. Konda, *Magellan: Toward building entity matching management systems.* The University of Wisconsin-Madison, 2018.

[91] Y. Zhu *et al.*, "A comprehensive study of deep video action recognition," *arXiv preprint arXiv:2012.06567*, 2020.

[92] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, *Bert: Pre-training of deep bidirectional transformers for language understanding*, 2019. arXiv: 1810.04805 [cs.CL].

[93] Y. Kong and Y. Fu, "Human action recognition and prediction: A survey," *International Journal of Computer Vision*, vol. 130, no. 5, pp. 1366–1401, 2022.

[94] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *Digital Signal Processing*, p. 103 514, 2022.

[95] Y.-C. Pu, W.-C. Du, C.-H. Huang, and C.-K. Lai, “Invariant feature extraction for 3d model retrieval: An adaptive approach using euclidean and topological metrics,” *Computers & Mathematics with Applications*, vol. 64, no. 5, pp. 1217–1225, 2012.

[96] R. Gasser, L. Rossetto, and H. Schuldt, “Multimodal multimedia retrieval with vitrivr,” in *Proceedings of the 2019 on International Conference on Multimedia Retrieval*, ser. ICMR ’19, Ottawa ON, Canada: Association for Computing Machinery, 2019, pp. 391–394, ISBN: 9781450367653. DOI: 10.1145/3323873.3326921. [Online]. Available: https://doi.org/10.1145/3323873.3326921.

[97] S. M. Sarwar and J. Allan, “Query by example for cross-lingual event retrieval,” in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR ’20, Virtual Event, China: Association for Computing Machinery, 2020, pp. 1601–1604, ISBN: 9781450380164. DOI: 10.1145/3397271.3401283. [Online]. Available: https://doi.org/10.1145/3397271.3401283.

[98] L. Zheng *et al.*, “Mars: A video benchmark for large-scale person re-identification,” in *European Conference on Computer Vision*, Springer, 2016, pp. 868–884.

[99] N. Rasiwasia *et al.*, “A new approach to cross-modal multimedia retrieval,” in *Proceedings of the 18th ACM international conference on Multimedia*, 2010, pp. 251–260.

[100] M. Kan, S. Shan, and X. Chen, “Multi-view deep network for cross-view classification,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4847–4855.

[101] Y. Peng, X. Huang, and J. Qi, “Cross-media shared representation by hierarchical learning with multiple deep networks,” in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, ser. IJCAI’16, New York, New York, USA: AAAI Press, 2016, pp. 3846–3853, ISBN: 9781577357704.

[102] F. Faghri, D. J. Fleet, J. R. Kiros, and S. Fidler, “Vse++: Improving visual-semantic embeddings with hard negatives,” *arXiv preprint arXiv:1707.05612*, 2017.

[103] X. Xu, L. He, H. Lu, L. Gao, and Y. Ji, “Deep adversarial metric learning for cross-modal retrieval,” *World Wide Web*, vol. 22, no. 2, pp. 657–672, 2019.

[104] J. Wei, X. Xu, Y. Yang, Y. Ji, Z. Wang, and H. T. Shen, “Universal weighting metric learning for cross-modal matching,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 13 005–13 014.

[105] M. Rohrmeier, “Towards a generative syntax of tonal harmony,” *Journal of Mathematics and Music*, vol. 5, no. 1, pp. 35–53, 2011.

[106] T. Osborne, *A dependency grammar of English: An introduction and beyond.* John Benjamins Publishing Company, 2019.

[107] R. J. Qureshi, J.-Y. Ramel, and H. Cardot, “Graph based shapes representation and recognition,” in *International Workshop on Graph-Based Representations in Pattern Recognition*, Springer, 2007, pp. 49–60.

[108] R. Socher, D. Chen, C. D. Manning, and A. Ng, “Reasoning with neural tensor networks for knowledge base completion,” in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2013, pp. 926–934. [Online]. Available: http://papers.nips.cc/paper/5028-reasoning-with-neural-tensor-networks-for-knowledge-base-completion.pdf.

[109] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” *CoRR*, vol. abs / 1609.02907, 2016. arXiv: 1609.02907. [Online]. Available: http://arxiv.org/abs/1609.02907.

[110] Z. Chen, A. Li, and Y. Wang, “Video-based pedestrian attribute recognition,” *CoRR*, vol. abs/1901.05742, 2019. arXiv: 1901.05742. [Online]. Available: http://arxiv.org/abs/1901.05742.

[111] N. McLaughlin, J. M. Del Rincon, and P. Miller, “Recurrent convolutional network for video-based person re-identification,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1325–1334.

[112] S. Ji, W. Xu, M. Yang, and K. Yu, “3d convolutional neural networks for human action recognition,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 221–231, 2012.

[113] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[114] V. E. Liong, J. Lu, Y.-P. Tan, and J. Zhou, “Deep coupled metric learning for cross-modal matching,” *IEEE Transactions on Multimedia*, vol. 19, no. 6, pp. 1234–1244, 2016.

[115] A. Frome, G. Corrado, J. Shlens, *et al.*, “Devise: A deep visual-semantic embedding model,” in *Advances in Neural Information Processing Systems*, vol. 26, 2013.

[116] K.-H. Lee, X. Chen, G. Hua, H. Hu, and X. He, “Stacked cross attention for image-text matching,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 201–216.

[117] Y. Song and M. Soleymani, “Polysemous visual-semantic embedding for cross-modal retrieval,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1979–1988.

[118] M. Lazaridis, A. Axenopoulos, D. Rafailidis, and P. Daras, “Multimedia search and retrieval using multimodal annotation propagation and indexing techniques,” *Signal Processing: Image Communication*, vol. 28, no. 4, pp. 351–367, 2013, ISSN: 0923-5965. DOI: https://doi.org/10.1016/j.image.2012.04.001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0923596512000756.

[119] X. Zhai, Y. Peng, and J. Xiao, “Learning cross-media joint representation with sparse and semisupervised regularization,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 6, pp. 965–978, 2013.

[120] G. Andrew, R. Arora, J. Bilmes, and K. Livescu, “Deep canonical correlation analysis,” in *International conference on machine learning*, PMLR, 2013, pp. 1247–1255.

[121] P. Hu, L. Zhen, D. Peng, and P. Liu, “Scalable deep multimodal learning for cross-modal retrieval,” in *Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR’19, Paris, France: ACM, 2019, pp. 635–644, ISBN: 978-1-4503-6172-9. DOI: 10.1145/3331184.3331213. [Online]. Available: http://doi.acm.org/10.1145/3331184.3331213.

[122] F. Feng, X. Wang, and R. Li, “Cross-modal retrieval with correspondence autoencoder,” in *Proceedings of the 22nd ACM international conference on Multimedia*, 2014, pp. 7–16.

[123] H. Luo *et al.*, “Univl: A unified video and language pre-training model for multimodal understanding and generation,” *arXiv preprint arXiv:2002.06353*, 2020.

[124] B. Wang, Y. Yang, X. Xu, A. Hanjalic, and H. T. Shen, “Adversarial cross-modal retrieval,” in *Proceedings of the 25th ACM international conference on Multimedia*, 2017, pp. 154–162.

[125] A. Bellet, A. Habrard, and M. Sebban, "A survey on metric learning for feature vectors and structured data," *arXiv preprint arXiv:1306.6709*, 2013.

[126] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[127] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 221–231, 2012.

[128] T. Boult *et al.*, "Towards a unifying framework for formal theories of novelty," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, 2021, pp. 15 047–15 052.

[129] B. Liu, E. Robertson, S. Grigsby, and S. Mazumder, *Self-initiated open world learning for autonomous ai agents*, 2021. arXiv: 2110.11385 [cs.AI].

[130] P. Langley, "Open-world learning for radically autonomous agents," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 13 539–13 543.

[131] H. Xu, B. Li, V. Ramanishka, L. Sigal, and K. Saenko, "Joint event detection and description in continuous video streams," *CoRR*, vol. abs/1802.10250, 2018. arXiv: 1802.10250. [Online]. Available: http://arxiv.org/abs/1802.10250.

[132] T. Mitchell *et al.*, "Never-ending learning," in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15)*, 2015.

[133] T. P. Tanon, G. Weikum, and F. Suchanek, "Yago 4: A reason-able knowledge base," in *European Semantic Web Conference*, Springer, 2020, pp. 583–596.

[134] R. Kiros *et al.*, "Skip-thought vectors," in *Advances in neural information processing systems*, 2015, pp. 3294–3302.

[135] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, *Distributed representations of words and phrases and their compositionality*, 2013. arXiv: 1310.4546 [cs.CL].

[136] K. Lang, "Newsweeder: Learning to filter netnews," in *Machine Learning Proceedings 1995*, Elsevier, 1995, pp. 331–339.

[137] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, vol. 41, no. 6, pp. 391–407, 1990.

[138] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, no. null, pp. 993–1022, Mar. 2003, ISSN: 1532-4435.

[139] A. Paszke *et al.*, "Automatic differentiation in pytorch," 2017.

[140] J. Duda, *Sgd momentum optimizer with step estimation by online parabola model*, 2019. arXiv: 1907.07063 [cs.LG].

[141] J. G. Moreno-Torres, T. Raeder, R. Alaiz-Rodríguez, N. V. Chawla, and F. Herrera, "A unifying view on dataset shift in classification," *Pattern Recognition*, vol. 45, no. 1, pp. 521–530, 2012, ISSN: 0031-3203. DOI: https://doi.org/10.1016/j.patcog.2011.06.019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031320311002901.

[142] J. Xu, T. Mei, T. Yao, and Y. Rui, "Msr-vtt: A large video description dataset for bridging video and language," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5288–5296.

[143] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. Carlos Niebles, "Dense-captioning events in videos," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 706–715.

[144] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[145] D. Angelov, "Top2vec: Distributed representations of topics," *arXiv preprint arXiv:2008.09470*, 2020.

[146] D. Wadden, U. Wennberg, Y. Luan, and H. Hajishirzi, "Entity, relation, and event extraction with contextualized span representations," *arXiv preprint arXiv:1909.03546*, 2019.

[147] K. Solaiman and B. Bhargava, "Measurement of novelty difficulty in monopoly," in *Proceedings of the AAAI Spring Symposium on Designing Artificial Intelligence for Open Worlds, Palo Alto, CA, USA*, Mar. 2022. [Online]. Available: https://usc-isi-i2.github.io/AAAI2022SS/papers/SSS-22_paper_65.pdf.

[148] S. Islam, K. Solaiman, R. De Oliveira, and B. Bhargava, “Domain complexity estimation for distributed ai systems in open-world perception domain,” DOI: 10.13140/RG.2.2.14853.63204.

VITA

KMA Solaiman received his B.Sc. in Computer Science and Engineering from Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh, in 2014, with a concentration on Artifical Intelligence. His B.Sc. dissertation titled ‘Minimal Parameter Clustering of Complex Shaped an Different Sized Dataset (MPCACS)’ focused on unsupervised learning. He completed his M.Sc. in Computer Science from Purdue University, West Lafayette, IN in Dec 2022. Solaiman served as a full time faculty member at Ahsanullah University of Science and Technology (AUST), Dhaka and United International University (UIU), Dhaka from 2014 to 2016 before moving to West Lafayette to pursue his Ph.D. degree in Computer Science at Purdue University, West Lafayette, IN, USA. During his time at Purdue, Solaiman has worked on REALM project funded by NGC and SAIL-ON project funded by DARPA.

He worked closely with Prof. Michael Stonebraker for the REALM/SKOD project and contributed in Multimodal Information Retrieval and Video and Text Feature Extraction. During the SAIL-ON project, he worked closely with Josh Alspector and rest of the NWG team on theories of novelty, including novelty detection, adaptation, and characterization in perception and planning domains.

His research interests revolve around multimodal data management with an interdisciplinary range of applications, spanning from data discovery to multimodal information retrieval, user modeling, explainable AI, natural language processing, and video feature extraction. He will be joining as a full time faculty member at University of Maryland, Baltimore County (UMBC) from Fall 2023.

PUBLICATIONS

1. S. Islam, K. Solaiman, R. Oliveira, B. Bhargava. Domain Complexity Estimation for Distributed AI Systems in Open-World Perception Domain. Submitted in **Artificial Intelligence, Open-World AI**, *July 2023*.

2. K. Solaiman and B. Bhargava. Multi-modal Information Retrieval for Systems with Explicit Information Needs and Object Properties (FemmIR), Submitted in **SIGMOD** *2023*.

3. Kma Solaiman, Tao Sun, Alina Nesen, Bharat Bhargava, and Michael Stonebraker. Applying Machine Learning and Data Fusion to the *Missing Person* Problem. ***IEEE Computer****, Volume: 55, Issue: 6, June 2022*.

4. K. Solaiman and B. Bhargava. Open-Learning Framework for Multi-modal Information Retrieval with Weakly Supervised Joint Embedding. In **AAAI Spring Symposium** on Designing Artificial Intelligence for Open Worlds, *March 2022*.

5. K. Solaiman and B. Bhargava. Measurement of Novelty Difficulty in Monopoly. In **AAAI Spring Symposium** on Designing Artificial Intelligence for Open Worlds, *March 2022*.

6. A. Nesen, K. Solaiman and B. Bhargava. Dataset Augmentation with Generated Novelties, **IEEE TransAI**, *2021*.

7. Michael Stonebraker, Bharat Bhargava, Michael Cafarella, Zachary Collins, Jenna McClellan, Aaron Sipser, Tao Sun, Alina Nesen, KMA Solaiman, Ganapathy Mani, Kevin Kochpatcharin, Pelin Angin, and James MacDonald. Surveillance Video Querying With A Human-in-the-Loop, In Workshop on Human-In-the-Loop Data Analytics **(HILDA)** with **SIGMOD**, *2020*.

8. S. Palacios, K. Solaiman, P. Angin, A. Nesen, B. Bhargava, Z. Collins, A. Sipser, M. Stonebraker. SKOD: A Framework for Situational Knowledge on

Demand, In Workshop on Heterogeneous Data Management, Polystores, and Analytics for Healthcare **(POLY)** at **VLDB**, *August 30, 2019.*

9. Kma Solaiman, MM Rahman, and N Shahriar. AVRA BANGLADESH: Collection, Analysis & Visualization of Road Accident Data in Bangladesh, **IEEE ICIEV**, *2013.*