2018 IEEE 38th International Conference on Distributed Computing Systems

Towards Distributed Cyberinfrastructure for Smart Cities using Big Data and Deep Learning Technologies

Shayan Shams †§, Sayan Goswami †§, Kisung Lee †, Seungwon Yang ‡§, and Seung-Jong Park †§
†Division of Computer Science and Engineering
‡School of Library and Information Science
§Center for Computation and Technology
Louisiana State University (LSU)
Baton Rouge, LA 70803, USA
sshams2@cct.lsu.edu, sgoswami@cct.lsu.edu, klee76@lsu.edu, seungwonyang@lsu.edu, sjpark@cct.lsu.edu

***Abstract*—Recent advances in big data and deep learning technologies have enabled researchers across many disciplines to gain new insight into large and complex data. For example, deep neural networks are being widely used to analyze various types of data including images, videos, texts, and time-series data. In another example, various disciplines such as sociology, social work, and criminology are analyzing crowd-sourced and online social network data using big data technologies to gain new insight from a plethora of data. Even though many different types of data are being generated and analyzed in various domains, the development of distributed city-level cyberinfrastructure for effectively integrating such data to generate more value and gain insights is still not well-addressed in the research literature. In this paper, we present our current efforts and ultimate vision to build distributed cyberinfrastructure which integrates big data and deep learning technologies with a variety of data for enhancing public safety and livability in cites. We also introduce several methodologies and applications that we are developing on top of the cyberinfrastructure to support diverse community stakeholders in cities.**

I. Introduction

According to a report published in 2016 by the United Nations, about 60% of the global population will live in cities by 2030, and about half of them will live in cities with half a million inhabitants or more [1]. An overwhelming portion of the urban population lives in mega-cities, which have a population of 10 million or more. As cities continue to grow, access to essential services such as health care, education, housing, transportation, and law enforcement becomes increasingly challenging. In particular, crime and traffic congestion become common and critical issues in many large cities. As the size of city population increases, the rates of violent crime and the traffic congestion index have soared while overall crime rates remain similar compared to previous years [2].

As the population of modern cities increases, the cities start to generate huge amounts of data from diverse sources such as IoT (Internet of Things) sensors, remote cameras, social media (e.g., Twitter), crowd-sourcing platforms (e.g., Waze), and interactive kiosks. These heterogeneous data are then shared and analyzed for better governance and efficient resource utilization for communities. Management and analysis of such a huge volume of structured and unstructured data requires not only cutting-edge hardware running state-of-the-art big data frameworks but also machine learning and artificial intelligence tools that are at the forefront of innovation.

For example, Chicago Mayor's Office and the Chicago Police Department (CPD) deployed new predictive technologies and analytical tools to reduce gun violence. After deploying a predictive platform equipped with integrated, geographic-specific, real-time analytics for crime data, video surveillance, and gunshot detection, one district in Chicago saw a nearly 60% reduction in the number of shooting incidents [3]. Such new data and technologies will drive modern cities into smart environments by enhancing public safety and livability of residents.

Among the new technologies, deep learning is one of the newest and fastest growing classes of new techniques, which has been under active development since its resurgence in 2006 [4]. Backed by the advancements in specialized hardware accelerators, deep learning has enabled various fields of study to analyze textual, multimedia, or network data, as well as gain more insights from such complex data. Although these fields are still being actively investigated by adopting the latest technologies, most current research efforts focus on individual problems, lacking a holistic and integrative approach.

While the transformation driven by new data and technologies reveals great promises, there still exist significant challenges to address for achieving improved well-being and prosperity in societies. One of the ways to tackle the challenges of a smart city is to integrate and understand information generated from different sources in a more holistic fashion. Furthermore, since deep learning models are extremely compute-intensive to train and data-intensive to run inferences on streaming videos and texts, we need an environment which juxtaposes big data technologies with traditional high-performance computing (HPC) augmented by state-of-the-art hardware accelerators.

In this paper, we present our current efforts and ultimate

2575-8411/18/$31.00 ©2018 IEEE
DOI 10.1109/ICDCS.2018.00127

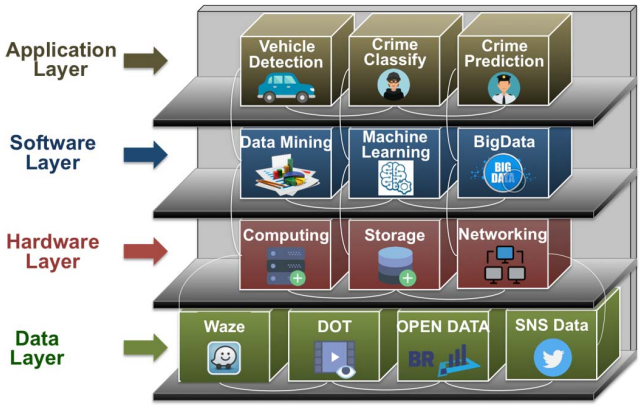

Fig. 1: An overview of our cyberinfrastructure.

vision to build distributed cyberinfrastructure that integrates big data and deep learning technologies with a variety of data for enhancing public safety and livability in cites. We also introduce several techniques and applications that we are developing on top of the cyberinfrastructure to support various community stakeholders in cities.

The rest of the paper is organized as follows. In Section II, we introduce an overview of our cyberinfrastructure which consists of multiple layers and components. Next, we describe our methodologies in Section III and applications that we are developing on top of the cyberinfrastructure in Section IV. Lastly, we conclude with our future research directions in Section V.

II. Cyberinfrastructure Overview

The architecture of our cyberinfrastructure consists of four layers as shown in Fig. 1. The data layer has multiple types of structured and unstructured raw data which we analyze, annotate, and index for data mining and visualization purposes. The data types include traffic and surveillance videos, crowd-sourced traffic reports, social network data, and publicly-available city data (e.g., crimes, traffic, emergency calls). The hardware layer consists of components for gathering our raw data, training and running our analytical models, and storing transformed data for future analysis. This layer includes edge devices which are equipped with sensors and cameras, temporary storage servers for raw data, analysis servers for training and running our models, and long-term storage servers for annotated data. The software layer consists of software tools and frameworks which we use to manage and analyze our stored data. This layer combines big data tools for streaming and storing huge amounts of raw data, deep learning frameworks to train models and run inferences for annotating the raw data, and visualization tools. Lastly, the application layer includes our real-world applications and services which we are building and planning to develop for the smart city initiatives. The bottom three layers are described in detail in the following subsections while we introduce the application layer in Section IV.

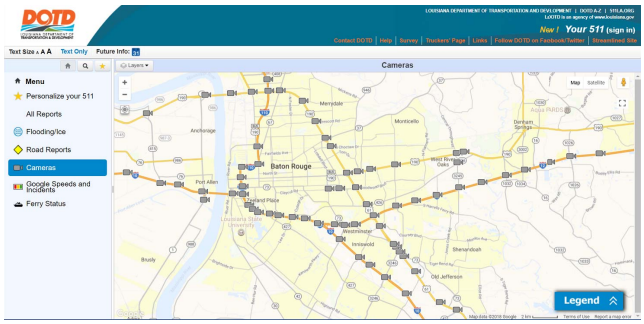

Fig. 2: DOTD cameras in interstate highways around Baton Rouge area.

A. Data Layer

Our cyberinfrastructure is designed to collect and manage heterogeneous types of data generated from various devices and domains. We currently focus on four types including videos, social network data, publicly-available city data, and law enforcement data.

1) Traffic and Surveillance Videos: Our cyberinfrastructure receives live video feeds from cameras of the Louisiana Department of Transportation and Development (DOTD) and the City of Baton Rouge for conducting real-time traffic and public safety analyses, as well as for performing data transformation for future data analyses. The DOTD cameras are installed along the major interstate highways in Louisiana covering several big and mid-sized cities including New Orleans, Baton Rouge, Houma, Shreveport, Lafayette, North Shore, Lake Charles, Monroe, and Alexandria. By connecting to the DOTD network, our cyberinfrastructure can access more than 200 cameras, which constantly provide live feeds from the highways across the state of Louisiana. Fig. 2 shows the locations of the DOTD cameras in Baton Rouge, the capital city of Louisiana.

2) Online Social Networks and Crowd-Sourced Traffic Reports: Online social networks, such as Twitter and Facebook, provide invaluable information not only for scientists in academic disciplines but also for government officials in law enforcement and homeland security fields. Using a cluster of machines, our cyberinfrastructure collects tweets via Twitter API based on specific keywords and geospatial coordinates. Users can easily add new keywords and locations to gather tweets of interest. Moreover, through a collaboration with social, environmental, and political scientists, our cyberinfrastructure stores comprehensive tweet datasets for some major natural disasters. In addition to online social network data, our cyberinfrastructure collects data from Waze, the world's largest crowdsourcing-based traffic and navigation application, to help city officials make better decisions in terms of the city traffic management and emergency responses. Through the Waze's Connected Citizens Program (CCP), we store and analyze real-time traffic information including system-generated traffic jams and user-reported traffic incidents.

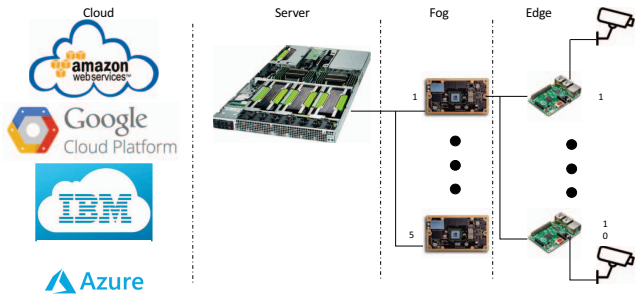

Fig. 3: The proposed video analysis pipeline dividing computation into multiple steps (Edge, Fog, Server, and Cloud) to provide fast and distributed analysis.

3) Publicly-available City Data: Open data for cities has become an important trend for improved transparency and potentially transformative data analytics [5]. The city of Baton Rouge, with which we are collaborating to build a smarter city, provides the data. This 'open' data is comprised of, among others, public safety information (e.g., crime incidents, fire incidents, medical clinics), housing and development information (e.g., census demographics, building permits), government information (e.g., citizen requests for services, public facilities, purchase orders), and transportation and infrastructure information (e.g., traffic incidents, potholes, traffic signals). As new types of data become available, those will also be incorporated into the process.

4) Law Enforcement Data: In addition to the publicly-available data, our cyberinfrastructure analyzes law enforcement data which may include sensitive information. A memorandum of understanding between Louisiana State University (LSU) and law enforcement agencies in Baton Rouge has allowed monthly transfer of individual-level violent crime data, including data on homicides, robberies, aggravated assaults, and illegal use of a weapon, from law enforcement agencies. This data is provided every month, and it includes incident report numbers, offense description, Louisiana criminal offense code, report address, offense district, date and time of the offense, law enforcement agency responsible for the report, and the names and demographic information on all persons involved (both victims and suspects) including home address and role in the incident. The crime data are uploaded to a secure web server in the LSU campus through a unique URL address by agencies on the first day of each month. Files uploaded to the secure web server are deleted after 90 days.

B. Hardware Layer

1) Computing: Our cyberinfrastructure is based on a fog computing model consisting of four tiers as shown in Fig. 3. The lowest tier is made up of edge devices, such as smartphones and Raspberry PIs (credit card-sized computers), that are responsible for collecting data from sensors and cameras and sending them upstream. Since these edge devices possess network connectivity and usually contain limited computation power and storage capacity, they can be used to perform elementary data filtering to reduce network communication to the higher tiers. They also act as buffers when transferring data from stateless devices to long-term storage servers in the cloud.

The next tier consists of fog nodes that are embedded devices such as NVIDIA Jetson. Each of these devices is responsible for aggregating data from a set of edge devices and sending them upstream for further analysis and storage. Since fog nodes are more powerful than edge devices, they can perform more advanced operations on the raw data. For example, we utilize fog nodes to run inferences using the first few layers of a deep learning model. When the fog nodes are confident about their inference results, only the annotated data is sent upstream for long-term storage and data mining. Otherwise, the raw data is sent upstream for further analysis.

The third tier includes analysis servers that are standalone nodes with modest to high processing power and in charge of handling compute-intensive tasks such as training deep learning models and running inferences on raw data using all layers of the trained model. Each analysis server handles a set of fog nodes and receives data from them in one of two forms: 1) the raw data when the fog nodes are not confident about their preprocessing results and 2) the data annotated by the fog nodes. After analysis on the analysis servers, the data is sent upstream for further processing.

The top tier is a federated cloud that consists of public cloud services (e.g., Amazon Web Services, Microsoft Azure, IBM Cloud) and open research infrastructures (e.g., GENI (Global Environment for Network Innovations), XSEDE (Extreme Science and Engineering Discovery Environment), Emulab). This tier fetches annotated data from analysis servers and stores it in distributed file systems (or database systems) for large-scale data mining and visualization.

2) Storage: The federated cloud in the top tier of our cyberinfrastructure contains short- and long-term storage subsystems in the public clouds as well as HPC data centers. These subsystems can be in the form of a distributed file system (e.g., Lustre, Hadoop Distributed File System (HDFS)) or a distributed NoSQL database system (e.g., HBase, MongoDB).

3) Networking: The federated cloud in the top tier of our cyberinfrastructure is connected to the other three tiers by Internet2, a member-owned nationwide high-speed network backbone consisting of educational and research institutions, government agencies, and leading corporations. Moreover, the three bottom tiers are interconnected by high-speed regional networks such as Louisiana Optical Network Infrastructure (LONI), which is available for use by academic, government, and industry partners in collaboration with LONI participant institutions.

C. Software Layer

1) Deep Learning: Recent remarkable advances in deep learning, exemplified with excellent success in image recognition, speech recognition, and natural language processing, have garnered a great deal of interest for a wide range of applications [6], [7], [8]. Backed by the advancements in special-

ized hardware accelerators [9], deep learning techniques have continued to thrive in various domains and disciplines [10], [11], [12], [13], [14]. To support deep learning-based analytics for smart city applications, our cyberinfrastructure uses TensorFlow because it provides model and data parallelism and can be easily distributed among multiple nodes and multiple workers per node.

2) Big Data Management: To store and analyze the large-scale data generated from various sources in a scalable and efficient manner, our cyberinfrastructure integrates various big data frameworks. First, we use Hadoop Distributed File System (HDFS), a distributed file system running on top of a cluster of machines, to store large-scale datasets. HDFS provides reliability and availability by replicating data blocks across multiple machines so, even though some machines may fail, we can still access the data stored in HDFS. On top of HDFS, we use Apache Hadoop YARN and Apache Spark as the resource scheduler and distributed data processing engine respectively. In addition, to gather data from legacy database systems, we utilize Apache Sqoop, a data import tool for bulk data transfers between RDBMSs (relational database management systems) and HDFS. For real-time data gathering, we use Apache Flume, a data import tool for real-time data transfers from various information sources.

For efficient query processing from huge amounts of heterogeneous data, we utilize various types of NoSQL database systems including HBase and MongoDB. Apache HBase is a distributed NoSQL database system running on top of HDFS. We can categorize HBase as a wide-column store or two-dimensional key/value store. Unlike HDFS that is optimized only for batch-style data access, HBase supports efficient random read/write operations. MongoDB is a document-based NoSQL database system optimized for storing unstructured or semi-structured documents such as JSON data. MongoDB is equipped with various indexing techniques for efficient query processing on various data types. Our cyberinfrastructure also supports other types of analytical workloads such as streaming processing, geospatial processing, and graph-based processing [15], [16], [17], [18].

3) Data Mining and Visualization: In addition to deep learning-based analytics, our cyberinfrastructure provides traditional machine learning and data mining capability for structured and annotated data. We utilize various distributed data mining tools including Apache Spark MLlib. Moreover, our cyberinfrastructure provides visualization capability for displaying both raw and analyzed data interactively. We currently utilize the D3 JavaScript visualization library for visualizing our data.

Fig. 4 shows the overall pipeline architecture in our cyberinfrastructure for data collection, management, analysis, and visualization. The raw input data are collected from multiple sources and stored in NoSQL databases for analysis in analysis servers. Analysis servers run different deep learning model for inference and the result of inference will be sent to the web server to be visualized on our website.

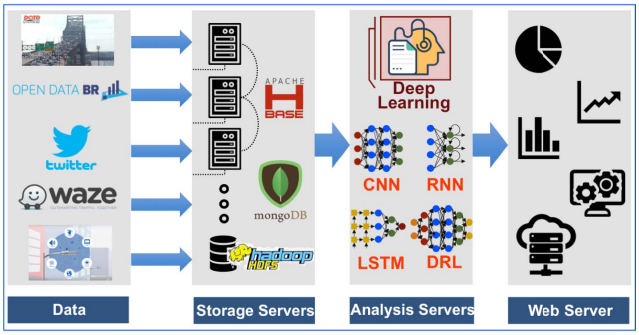

Fig. 4: Overall pipeline for data collection, management, analysis, and visualization.

III. Methodologies

In this section, we explain our methodologies that we use in developing our analysis models on top of the cyberinfrastructure. Our methodologies include a number of machine learning, in particular, deep neural network modules. The modules serve as building blocks that can be configured and combined together to construct our analysis models. The models are then used in the application layer to perform various analysis tasks, as described in Section IV.

A. Spatial Analysis

In spatial analysis, we aim to discover spatial patterns in the data such as patterns in images and geospatial patterns (e.g., patterns of criminal activity locations). We develop a collection of Convolutional Neural Network (CNN) modules for spatial analysis in our cyberinfrastructure. Through the convolutional operation, a neuron in a CNN generates responses to local spatial patterns at different locations. The set of activities produced by the neuron scanning through the image gives rise to a feature map. A CNN may consist of multiple layers, where the feature maps from a lower layer serve as input to a higher layer. Such a network forms a hierarchy, in which large spatial patterns (i.e., receptive fields of neurons at higher layers) can be constructed from smaller one (i.e., receptive fields of those in the lower layers). CNN-based networks have shown great success in many image processing tasks, from classification and object recognition to medical image diagnosis. Our CNN modules enable highly effective analysis in our cyberinfrastructure.

We can also conduct spatial analysis using CNN beyond image data. For example, in DeepMind's AlphaGo, CNN is used to analyze the positions of the stones on the GO game board. There are plenty of scenarios in our cyberinfrastructure that deal with geospatial data. Examples include traffic congestion, criminal activities, and economic development levels at different locations. Such data can be viewed as geospatial "images" and analyzed using CNNs. Our collection of CNN modules includes several CNN variants. Besides the regular CNNs, we also include inception types of CNN as used in the GoogleNet and the ResNet type of CNN. The variants will

allow the applications to try and employ the CNNs that are most suitable for the scenarios.

B. Temporal Analysis

Temporal analysis is the process of analyzing and discovering temporal patterns in the data. To support temporal analysis in our cyberinfrastructure, we develop a collection of Recurrent Neural Network (RNN) modules. Many researchers have shown success using RNNs on different types of sequential data, from natural language understanding to action recognition. A specific type of RNNs that we include is the Long-Short-Term Memory (LSTM) network. LSTM's capability of discovering long-range correlations is particularly useful for time series. Our application layer includes behavior analysis, which involves patterns of actions along time. LSTM networks serve as important modules in such an analysis. The RNN modules can also be used for analyzing sequential data such as texts which consist of sequences of words. Our cyberinfrastructure can deal with large amounts of text data from social media and other sources. Our RNN modules will enable effective text processing.

C. Multi-Modal Analysis

Community challenges are often perpetuated by the large number of disconnected sources of information from both people and technology. To address the challenges, we develop multi-modal analysis that learns features and makes decisions by combining and fusing information of multiple modals, such as video (image data) and sound (audio data) for gun shots. It has been demonstrated in many learning situations that combining data from multiple modals can greatly increase the performance of a learning system [10]. Our cyberinfrastructure includes components for multi-modal fusion and analysis. One approach we implement is fusion based on deep auto-encoders. The encoders generate features for combining information from multiple channels. Another analysis that we include is canonical correlation analysis (CCA) [19].

D. Deep Reinforcement Learning (DRL)

Reinforcement Learning (RL), the paradigm of learning by trial-and-error, has been an essential framework for robotics, control systems, and AI research domains for many decades. Overcoming previous limitations for RL applications, mostly associated with fundamental challenges in high-dimensional data and model complexity, DRL opens a way to solve major roadblocks for its potential for a diverse set of AI applications [20], [21], [22], [23]. By leveraging the recent advances in DRL, human-level video tracking and incident reasoning, combinations of audio and video signals for control systems, and sequential decision-making systems using human perception data (e.g., audio, video, text) are achievable and can be more advanced. Our cyberinfrastructure includes components for deep reinforcement learning to develop various smart city applications, such as smart camera controls to automatically rotate and zoom in for traffic and crime incidents.

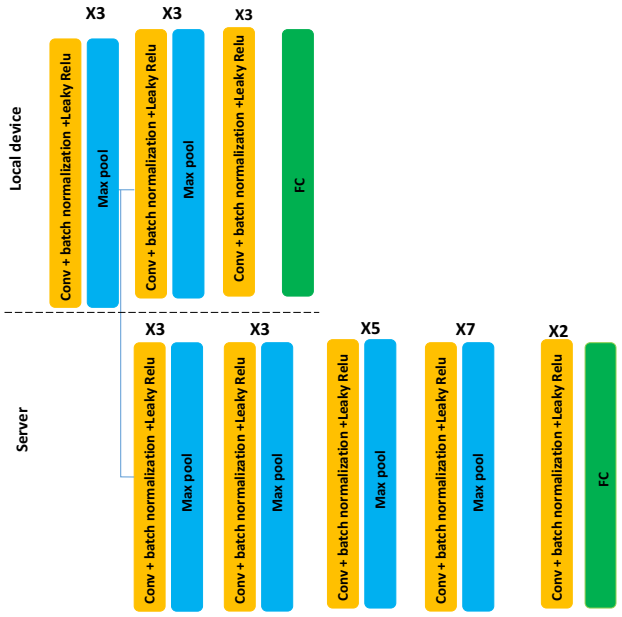

Fig. 5: Deep learning architecture for vehicle detection and classification. The architecture includes the tiny Yolo and Yolo models running on a local device and a server respectively. If the prediction accuracy using the tiny Yolo model is less than the predefined threshold, the feature map (shown with blue line) will be sent to the server for in-depth analysis.

IV. Applications

In this section, we introduce our applications that we are developing on top of the cyberinfrastructure using the methodologies explained in Section III.

A. Video Analysis-Based Applications

Compared to image data, video data contain richer types of information for objects, such as their contextual time-series information. Therefore, analyzing video data can not only give us the more accurate reconstruction of past events such as crime and traffic incidents but also allow us to predict critical events in the near future. Despite many potential benefits from video analysis, it is known to be difficult because of the huge size and complexity of streaming video data. By tackling the challenge based on the fog computing model described in Section II, we are developing various types of video analysis-based smart city applications as follows.

1) Vehicle Detection and Classification: Identifying details of vehicles (e.g., make, model, year, color) from video streams can be critical when tracking cars that are involved in criminal activities (e.g., tracking cars described in AMBER Alerts). Such analysis is time-consuming and error-prone if manually conducted by humans. To address this challenge, we are developing an application with a deep learning model, which can detect cars and classify them into the detailed vehicle information, from streaming video data. We use the Stanford car dataset [24] and our own crawled images from Google

Fig. 6: Examples of vehicle detection and classification based on the proposed deep learning model shown in Fig 5.

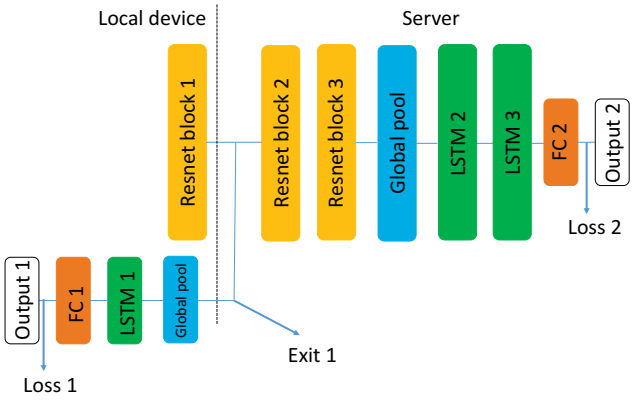

Fig. 7: Deep learning architecture for suspicious behavior and crime action recognition. The architecture includes two computation paths running on local device and server respectively. If the prediction accuracy using model running on the local device (exit path 1) is less than the predefined threshold, the feature map from Resnet block 1, will be sent to the server for in-depth analysis.

for common car models in the United States to train our current model. The combined dataset has 32,000 images for 400 classes. In our current prototype system, we use the Tiny YOLO (You Only Look Once) and YOLOv2 models [25] because YOLO provides fast and accurate detection and classification. We first run Tiny YOLO on local devices (i.e., edge devices or fog nodes). If the score of the classification is higher than a predefined threshold, the output of Tiny YOLO is considered as an acceptable outcome. Otherwise, the feature map obtained before the branch is sent to the analysis server in which it goes through the remaining YOLOv2 layers to obtain the object's bounding box and its predicted class. Fig. 5 shows our prototype deep learning architecture for vehicle detection and classification, divided between the local device and the analysis server. Fig. 6 demonstrates the vehicle detection and classification results using our prototype system.

2) Suspicious Behavior and Crime Action Recognition: Recognizing abnormal (or concerned) human actions and

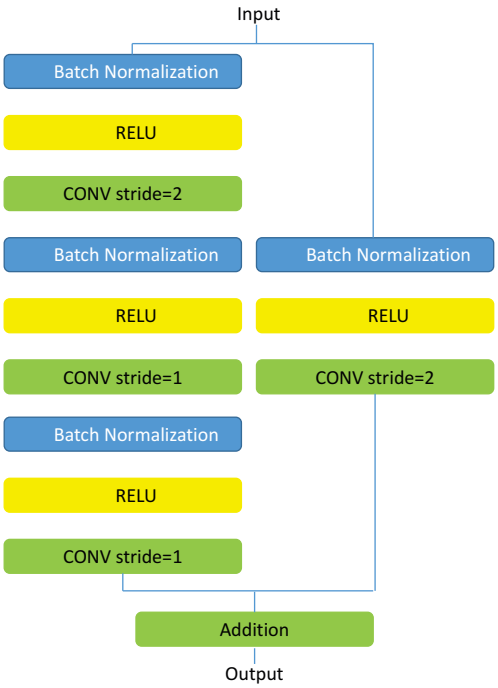

Fig. 8: ResNet block architecture illustrated in Fig. 7 For suspicious behavior and crime action recognition model. In our implementation, we use a convolutional layer for shortcut path instead of max pooling layer mostly used in Resnet block architecture.

behaviors from surveillance video streams are critical for fighting against crimes in smart cities considering that these techniques could be used for detecting and predicting crime events (e.g., jaywalking, hit-and-run events, armed robberies). Provided that human actions and behaviors involve a time dimension, we are developing an application for action and behavior recognition by combining both CNN and RNN modules. Fig. 7 shows our prototype deep learning architecture for suspicious behavior and crime action recognition. Our architecture integrates a local device with an analysis server to ensure highly-accurate and robust crime action recognition. Our CNN module, a stack of multiple ResNet [26] blocks, is responsible for analyzing activities within each frame in a video stream and transform a frame into a representation of activity features. Fig. 8 shows the details of our current ResNet block architecture used in this application. At each time step t, the CNN module processes the frame with time stamp t and outputs a representation for that frame. The sequence of the CNN's outputs along time will serve as the input to the RNN module, which consists of multiple LSTM layers. The LSTM layers extract temporal patterns along the per-time step activity representations. A final classifier, composed of one or more fully connected layers, takes the temporal patterns and generates recognition decisions.

To run the application based on the fog computing model described in Section II, we first execute ResNet block 1 with

LSTM 1 and FC1 on the local device (i.e., edge devices or fog nodes). If the entropy score of Output 1 (i.e., classification result) is higher than a predefined threshold, we index the video using Output 1 on the local device. Otherwise, the feature map obtained by ResNet block 1 is sent to the analysis server in which it goes through the remaining network to obtain Output 2. We then index the video using Output 2 on the analysis server.

We train the model using previously recorded videos from the city's street and traffic cameras. We split the videos into clips of several minutes in length and label them into different categories of suspicious behaviors and crime activities with the help of experts. Once the model is trained, it can be deployed to monitor video streams from several street and traffic cameras. When a suspicious behavior or crime activity is recognized, our application will log the time, location, the type of activity, and the video feed during that time window into a database. An alert will be sent to a human operator who reviews the information and the original video clip, and forwards the information to the authorities if needed.

B. Social Network Analysis-Based Applications

Social network analytic techniques are used by law enforcement to identify social relationships which interconnect violent offenders and criminal group members. By uncovering the social connections of a victim or a suspect, law enforcement may focus their investigations on individuals who are known to have a relationship history (either through conflict or collaboration). Social network relationships are detected by identifying the first-degree associates, individuals who are linked in place and time through criminal incident reports and/or through known gang or group affiliations. While this approach is useful for monitoring violent group and gang-related activity, and can be proved to be successful for uncovering leads to informants, witnesses, and co-offenders involved in criminal events, the sphere of social connectivity can be exhaustively large for timely analyses and investigations.

For example, of the 67 groups and gangs and their 982 members identified and observed in Baton Rouge area over the past 6 years, each gang member has a network size of 14 first-degree associates on average. However, best-practices suggest that investigative techniques extend to second-degree affiliates as well (i.e., a relationship connection through a shared co-offender). This approach may yield a field of interest which contains approximately 200 second-degree associates. These numbers are prohibitively large for an under-resourced staff of law enforcement agencies to investigate on a regular basis.

To address this challenge, we are developing a deep hybrid model which captures the temporal and textual features of criminal and gang networks constructed by Twitter data of the criminals and gang members, along with deep learning and natural language processing (NLP) techniques. First, we identify the Twitter IDs of the first- and second-degree associates of criminal and gang members. Next, we use NLP techniques to capture textual features present in tweet text at given times and locations associated with violent criminal incidents. Using a multi-modal algorithm, we integrate different types of information to determine whether a tweet from a criminal or gang associate falls within the specified time and location field of interest. The advantages of using multi-modal model are triangulation of event locations, times, and social relationships across complex and extensive volumes of data.

By combining the expansive field of second-degree associates with geo-targeted tweets during the time frame of a violent incident, the field of associates may be strategically narrowed to known associates who might have been in the location of a criminal incident at the time of the event. This layering of data may provide a tighter focus around a much smaller persons-of-interest field for detail investigations, and thus it may result in a more efficient use of law enforcement resources.

V. Conclusion

In this paper, we presented our current efforts and ultimate vision to build the distributed cyberinfrastructure that integrates big data and deep learning technologies with a variety of data for enhancing public safety and livability in cites. We also introduced several methodologies and applications that we have been developing on top of the cyberinfrastructure to support diverse community stakeholders in cities.

In addition to the aforementioned methodologies and applications, we plan to continuously extend our cyberinfrastructure in close collaboration with community stakeholders to conduct integrative research that transforms existing distributed computing capabilities in enhancing community well-being. One of our future research directions is integrating health care-related data, such as medical history and radiology images, into our cyberinfrastructure to support the transformation of health and medicine in cities. Through a MOU (Memorandum of Understanding) between LSU and several medical schools and centers in Louisiana, we started collecting anonymized medical data into our cyberinfrastructure. In this integrative research, we aim to address various types of challenges including not only technical challenges such as big medical data management and scalable distributed computing but also legal and ethical challenges such as HIPAA (Health Insurance Portability and Accountability Act)-compliant data storage and processing.

One critical health care-related problem that we are particularly interested in is the opioid epidemic in the United States. According to the U.S. Department of Health and Human Services, 116 people died every day from opioid-related drug overdoses in 2016 [27]. Deep learning-based analytics using our cyberinfrastructure may uncover additional factors that explain why opioid mortality rates are at epidemic levels. Data sources that we plan to analyze include, but not limited to, social network analysis, online social networks (e.g., Twitter, Facebook, etc.), the number of opioid prescriptions in Baton Rouge, traffic volume/DOTD data, drug-related activities in community, substance use-related crime arrests, locations of overdoses, 911 calls, and community knowledge (e.g., residents, law enforcement, coroner, etc.).

ACKNOWLEDGMENTS

This work was partially funded by NSF grants (MRI-1338051, IBSS-L-1620451, SCC-1737557, RAPID-1762600), NIH grants (P20GM103458-10, P30GM110760-03, P20GM103424), LA Board of Regents grants (LEQSF(2016-19)-RD-A-08 and ITRS), and IBM faculty awards.

REFERENCES

[1] Department of Economic and Social Affairs, Population Division, "The worlds cities in 2016," *Data Booklet (ST/ESA/ SER.A/392)*, 2016.

[2] The Brennan Center for Justice at NYU School of Law. (2016) Preliminary Analysis of 2015 FBI Uniform Crime Report. [Online]. Available: https://www.brennancenter.org/analysis/preliminary-analysis-2015-fbi-uniform-crime-report

[3] Mayor's Press Office. (2017) Mayor Emanuel Announces Expansion of Predictive Crime Strategy to Ogden. [Online]. Available: https://www.cityofchicago.org/city/en/depts/mayor/press_room/press_releases/2017/october/OgdenPredictiveTech.html

[4] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[5] M. S. Brown. (2017) Free Data Sources: Municipal Open Data Portals For 85 US Cities. [Online]. Available: https://www.forbes.com/sites/metabrown/2017/06/30/quick-links-to-municipal-open-data-portals-for-85-us-cities

[6] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[7] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85–117, 2015.

[8] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT Press, 2016.

[9] S. Shams, R. Platania, K. Lee, and S.-J. Park, "Evaluation of deep learning frameworks over different HPC architectures," in *Distributed Computing Systems (ICDCS), 2017 IEEE 37th International Conference on*. IEEE, 2017, pp. 1389–1396.

[10] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *Proceedings of the 28th international conference on machine learning (ICML-11)*, 2011, pp. 689–696.

[11] N. Ruchansky, S. Seo, and Y. Liu, "CSI: A Hybrid Deep Model for Fake News," *arXiv preprint arXiv:1703.06959*, 2017.

[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[13] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in neural information processing systems*, 2014, pp. 568–576.

[14] R. Platania, S. Shams, S. Yang, J. Zhang, K. Lee, and S.-J. Park, "Automated Breast Cancer Diagnosis Using Deep Learning and Region of Interest Detection (BC-DROID)," in *Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*. ACM, 2017, pp. 536–543.

[15] J. E. Gonzalez, R. S. Xin, A. Dave, D. Crankshaw, M. J. Franklin, and I. Stoica, "GraphX: Graph Processing in a Distributed Dataflow Framework," in *Proceedings of the 11th USENIX Conference on Operating Systems Design and Implementation*, ser. OSDI'14. Berkeley, CA, USA: USENIX Association, 2014, pp. 599–613. [Online]. Available: http://dl.acm.org/citation.cfm?id=2685048.2685096

[16] K. Lee, L. Liu, K. Schwan, C. Pu, Q. Zhang, Y. Zhou, E. Yigitoglu, and P. Yuan, "Scaling Iterative Graph Computations with GraphMap," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, ser. SC '15. New York, NY, USA: ACM, 2015, pp. 57:1–57:12. [Online]. Available: http://doi.acm.org/10.1145/2807591.2807604

[17] Y. Zhou, L. Liu, K. Lee, and Q. Zhang, "GraphTwist: Fast Iterative Graph Computation with Two-tier Optimizations," *Proc. VLDB Endow.*, vol. 8, no. 11, pp. 1262–1273, Jul. 2015. [Online]. Available: http://dx.doi.org/10.14778/2809974.2809987

[18] K. Lee, L. Liu, R. Ganti, M. Srivatsa, Q. Zhang, Y. Zhou, and Q. Wang, "Lightweight Indexing and Querying Services for Big Spatial Data," *IEEE Transactions on Services Computing*, 2016.

[19] A. Benton, H. Khayrallah, B. Gujral, D. Reisinger, S. Zhang, and R. Arora, "Deep generalized canonical correlation analysis," *arXiv preprint arXiv:1702.02519*, 2017.

[20] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[21] X. B. Peng, G. Berseth, and M. Van de Panne, "Terrain-adaptive locomotion skills using deep reinforcement learning," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, p. 81, 2016.

[22] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot *et al.*, "Mastering the game of Go with deep neural networks and tree search," *nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[23] J. Clark. (2015) Want machines to learn the way human toddlers do? You need a classroom equipped with Lego blocks and plenty of patience. [Online]. Available: https://www.bloomberg.com/features/2015-preschool-for-robots

[24] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3D Object Representations for Fine-Grained Categorization," in *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*, Sydney, Australia, 2013.

[25] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 779–788.

[26] 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.

[27] U.S. Department of Health and Human Services. (2017) About the U.S. Opioid Epidemic. [Online]. Available: https://www.hhs.gov/opioids/about-the-epidemic/