Generation and management of training data for AI-based algorithms targeted at coalition operations

Dinesh Verma, Greg Cirincione, Tien Pham, Bong Jun Ko

Dinesh Verma, Greg Cirincione, Tien Pham, Bong Jun Ko, "Generation and management of training data for AI-based algorithms targeted at coalition operations," Proc. SPIE 10635, Ground/Air Multisensor Interoperability, Integration, and Networking for Persistent ISR IX, 106350U (4 May 2018); doi: 10.1117/12.2305244

SPIE.

Event: SPIE Defense + Security, 2018, Orlando, Florida, United States

Generation and Management of Training Data for AI-based Algorithms Targeted at Coalition Operations

Dinesh Verma*[a], Greg Cirincione[b], Tien Pham[b], Bong Jun Ko[a]

[a]IBM T J. Watson Research Center, 1101 Kitchawan Road, Yorktown Heights, NY 10598, USA;
[b]U.S. Army Research Laboratory, 2800 Powder Mill Road, Adelphi, MD 20783, USA

ABSTRACT

AI (Artificial Intelligence)-based algorithms have great potential for inter-operation of coalition ISR (intelligence, surveillance, and reconnaissance) systems, but rely on realistic data for training and validation. Getting such data for coalition scenarios is hampered by military regulations and is a significant hurdle in conducting basic research. We discuss an approach whereby training data can be obtained by means of scenario-driven simulations, which result in traces for network devices, ISR sensors and other infrastructure components. This generated data can be used for both training and comparison of different AI based algorithms. Coupling the synthetic data generator with a data curation system further increases its applicability.

Keywords: Data generation, Training data for AI, Data management, Scenario-based data synthesis

1. INTRODUCTION

AI (Artificial Intelligence) algorithms based on machine learning (ML) techniques have a tremendous potential to improve the efficiency of ISR (intelligence, surveillance, and reconnaissance) systems' operation by automating the analysis of data acquired by various ISR sensors in order to gain operational insights. In a tactical coalition environment in particular, they enable a timely and effective information sharing between coalition partners without the need of exchanging raw sensor data, because only the essence of the data in the form of labels and contexts extracted from the analysis can be shared with the coalition partners. This has added benefits of effectively combating the communication bandwidth limitation in the battlefield situations as well as security and privacy concerns in the raw sensor data exchange.

Recent advances in deep learning algorithms and their application in a variety of data types (images, videos, texts, and time-series data) are increasingly making this potential a reality. One of the biggest hurdles in this prospect, however, is the lack of quality data sets required to effectively train the machine learning models. While this hurdle is present for developing deep learning models in all domains as it requires extensive efforts by human to collect, label, and curate the data set, the problem is especially difficult in tactical situations because operational data in tactical settings are almost invariably classified, making it difficult for the wider machine learning community of researchers and data scientists to develop, build, and refine the models relevant to the operational scenarios. The problem is specially acute for researchers who are working in fundamental research programs looking at military scenarios, e.g. researchers in the distributed analytics International Technology Alliance [1]

In this paper, we propose an approach to addressing the issue of data unavailability for developing machine learning models, present the architecture of a system that generates synthetic data, and discuss several methods for automatically generating realistic data for ISR operation. We note that, synthetically generating realistic data provides an additional benefit in coalition environment beyond effectively combating the lack of data: it enables an effective method for exchanging realistic (but not real) data across with coalition partners in a cross-domain resource sharing scenario, where the computation and storage resources of coalition partners can be utilized to perform computationally expensive tasks of processing the operational data, e.g., training deep neural network models.

Ground/Air Multisensor Interoperability, Integration, and Networking for Persistent ISR IX,
edited by Michael A. Kolodny, Dietrich M. Wiegmann, Tien Pham, Proc. of SPIE Vol. 10635,
106350U · © 2018 SPIE · CCC code: 0277-786X/18/$18 · doi: 10.1117/12.2305244

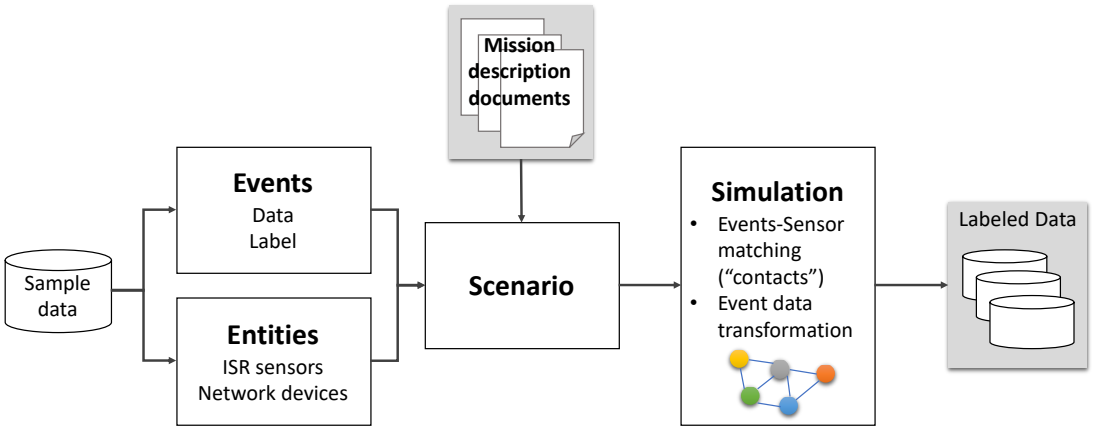

Figure 1. High-level view of the scenario-driven simulation architecture for data generation for ISR scenarios.

The basic premise of our approach is a scenario-driven simulation of ISR entities (sensors, devices, and environments) that are used in data acquisition activities and physical events that represent the physical phenomena as the target for the data acquisition in operational field. In our simulation framework, various entities and events are modeled and instantiated as autonomous elements, and the data acquisition of the raw event data by ISR sensors is modeled after physical characteristics of signal propagation from the source of the events and the measurement points. This way, we are able to capture (approximately) various event data observed by ISR sensors in specific operating conditions and contexts, as well as assign the labels to them with those of the ground-truth events. For generating the raw, ground-truth data of the targeted physical phenomena as well as the behaviors of the event sources and ISR entities, we also leverage on the recent advances in the use of deep generative models to synthesize realistic data set from samples.

One may argue that there are increasingly many datasets for machine learning, and that they can be selectively used to train ML models by identifying the data relevant to the specifics of ISR scenarios. This is true but only to a limited extent for two reasons. First, it has been observed in many practices that such a general dataset can be effectively used to build a baseline model, specifically for building feature extraction models, and then to use transfer learning techniques [2] to build the model for the target data domain with smaller dataset. However, such method will find little success if the size of the data in the target domain is extremely small or even non-existent due to regulatory reasons with respect to the complexity of the situation to be analyzed. Second, it is very difficult to selectively find the data relevant to operational environments of tactical situation which are suitable for handling scenarios of interest to coalition operations. For example, large public databases of labeled image data such as ImageNet [3] are unlikely to be useful in building a ML model that can detect and classify the objects and the context of the scene within the images for the particular context of active battlefield situation in an urban environment.

2. SCENARIO-DRIVEN SIMULATION FOR ISR DATA GENERATION

In this section, we present an architecture of the scenario-driven simulation system for ISR data generation,and describe the individual components in the architecture. Figure 1 illustrates a high-level architecture of the data generation framework.

2.1 Entities

Entities refer to the operational elements involved in ISR operations, including ISR sensors and other infrastructure devices (e.g., network, compute and storage devices). An entity is defined by its capabilities, inputs it takes, and outputs it generates. Table 1 lists examples of entities and their definitions.

Note that the example sensors in Table 1 include mobile sensors (e.g., aerial camera), implicitly indicating that they are instrumented in some mobile entities like unmanned aerial vehicles (UAVs) or as body-cams. An alternative way is to define them all as static entities, yet associate them (i.e.., "mount" them) with mobile units (e.g., UAVs, soldiers, etc.) defined as infrastructure entities in the scenario definition. This way, multiple sensors can be co-located (or move together in a unit) in the simulation.

A critical aspect of the entity definition is an approach to specify their behaviors. In our framework, the entities are defined as autonomous elements in that their behaviors are self-defined. For example, when an "instance" of an entity is created, its behaviors, such as mobility, directionality, sampling rate, etc., are pre-defined in the form of either deterministic parameters/traces, statistical models, or finite state machines. When it comes to performing the simulation,

Table 1. Examples of entities.

Entity name	Type	Capabilities	Communication
Aerial camera	Sensor	Modality: Images Mobile Max speed	Wifi Satellite
Surveillance camera	Sensor	Modality: Videos Static Coverage	Wifi
Acoustic array	Sensor	Modality: sound Coverage Mobile/Stationary Directional/Omni-directional	Wifi Satellite
Radar	Sensor	Modality: radar Coverage Stationary	Wifi
Wifi hotspots	Infrastructure	Mobile/Stationary Coverage	Wifi
Data sink	Infrastructure	Storage Compute	Wifi Satellite

these behaviors are exposed as externally visible parameters, so that the interaction between the entities and their recognition of the events can be efficiently simulated via matching them with those of the events.

2.2 Events

Events are the physical phenomena occurring in time and space within a certain environment; for example, the occurrence of explosive sounds, visual appearance and disappearance of physical entities like persons and vehicles, or simply some static information such as terrain data. They are the targets of data acquisition by the entities (i.e., captured by ISR sensors and collected by the infrastructure entities), and defined in a way similar to how the entities are defined, through parameters defining behaviors such as mobility, detectable coverage, fidelity, etc. These events are presented either in the form of the real data (sample data or data from other available data sets), or generated through data synthesis; We will discuss this event generation issue in further details in Section 4.

For our purpose of generating the training data for machine learning, these events are annotated with ground-truth *labels* that specify their nature—they constitute the "labels" part of the training data set. The "data" part is what is actually captured by the ISR sensors and collected through infrastructure entities. For a more realistic data generation, the quality of the captured event data in simulation should be varied depending on multiple factors. For example, a visual image of a target vehicle captured by a ground camera is degraded to a low quality in resolution, or a sound data contains a high-level of noise, due to a large distance between the target and the sensor, and/or due to the limited communication bandwidth from the sensor to the data sink. Note that some of the events may well be completely missed if their behaviors (i.e., occurrence in time and space) cannot be captured by the sensors (or "matched" with the sensors behaviors), in which case such event data will not be included in the training data set.

2.3 Scenario

A scenario defines the deployment of different entities and the occurrence of the events in time and space of a certain geographic environments. This is done by setting the parameters of the entities and events, specifying the location (or initial location in the case of mobile elements) and the times of appearance and disappearance. The actual environment they are deployed in consists of two parts: (i) static elements, such as the specification of a certain geographic area, and (ii) dynamic elements, such as conditions on visibility (e.g., due to weather) and communication (e.g., due to noises). These environment setting can be either synthetic (e.g., an open plain with a certain visual/radio propagation patterns) or

realistic (e.g., actual latitude-longitude coordinates with the actual weather condition). The spectrum of the environmental settings present the tradeoff between the complexity and the realism of the data generation: the synthetic and model-based setup requires less computation in the simulation, while the realistic ones present the opportunity to generate more realistic data (e.g., from the aerial images of an actual geographic region). With the modular design of the entities, events, and environments, our simulation architecture should allow "plugging in" different instances of the environmental settings and hence exploring broad spectrum of the data generation.

To facilitate the simulation, a scenario is specified in a structured format, that is, a well-defined set of the entities/events/environments, and their parameters defined in a structured language, such as XML (eXtensible Markup Language). The most straightforward method of generating a scenario would be by definition of human domain experts, who understands the requirements and environments of typical ISR operations. More advanced methods would be to generate the scenario in semi-automatic or automatic ways from corpus of natural language texts describing the operational scenario. We will discuss the possibility of the latter methods in Section 4.

2.4 Simulation

A simulation is carried out from the scenario definition. The most essential element of the simulation is the concept of *contacts* between the sensors and the events. A contact defines the moment and duration at/in which an event is observable (or is "covered"[4]) by a sensor during the simulation. More formally, a contact is made between an acoustic event E occurring at time t and a sensor S if

- E is within the sensing coverage of S at time t, and
- S is within the sensible range of E at time t.

This defines a point contact in time, while a sustained contact can be defined in a similar way over a time period $[t_1, t_2]$. Furthermore, each contact is associated with a *quality* transformation vector, which indicates the degradation in the quality of the event data when it is observed by the sensor. For example, an event of an explosion contacts an acoustic array sensor when it is heard by the sensor as both are within the range of each other, and the quality of the sound is degraded while it propagates to the sensor in the form of reduced magnitude and added environmental noise (or in other words, increased signal-to-noise ratio). Additionally, another type of contact is defined between a sensor S and a data collection entity C in a similar manner, when S is able to send the data measured at time t to C through infrastructure nodes at some point $t' > t$, possibly with degradation in data quality due to congestion and communication bandwidth limitation.

It is noteworthy that such contacts can be effectively represented by a time-varying graph, in which the nodes are events and entities and the edges denote the contacts between them, annotated by the contact time and the quality transformation vector. The process of the simulation is therefore basically to generate this time-varying graph based on the specified scenario, which in turn includes the definitions and deployments of the events, entities, and environments. The final outcome of the simulation is then produced via transforming the original, ground-truth event data into the measured/collected data according to this time-varying graph, in the form of a collection of tuple $<t, l, d, t', d'>$, where t denotes the time of event, l the ground-truth label of event, d the original event data, t' the time of measurement (or collection), and d' the measured/collected data about the event.

Note that, for a given event data $<t, l, d>$, there can be multiple measurements $<t', d'>$ when the event is observed by more than one sensors. Generating such multitudes of the measurements of the same event will be useful in building a ML model for multi-sensor analysis. Note also that we include the time index t in the collected data set because it will be of importance for sequential analysis of the data.

3. RAW DATA GENERATION

In this section, we discuss several methods for generating the raw data for events, entities, and scenarios, that are used to drive the simulation-based realistic dataset generation for ISR scenarios. Using synthetic data as the workload to drive the simulation has the benefit of not having to expose the real data subject to regulatory and security restriction. The methods we discuss here are largely based on the recent advances of machine learning techniques, in particular on generating realistic data from sample data sets in an unsupervised manner. The application of such techniques are different for the specific type of the raw data to be generated.

3.1 Synthesizing Raw Data from Samples

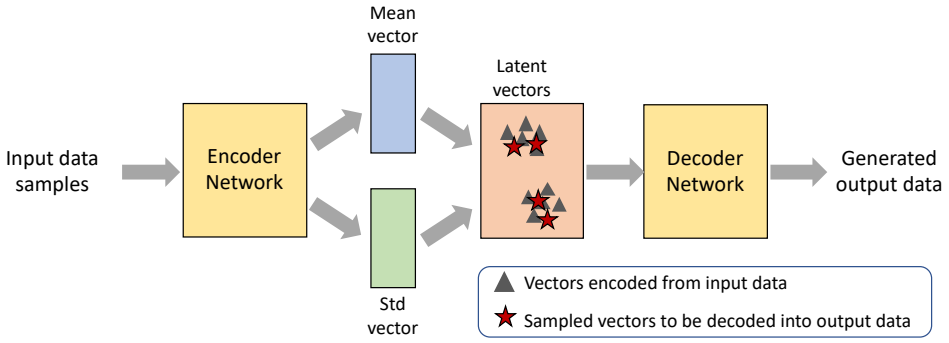

Figure 2. Raw data generation from sample data via VAE (Variational Auto-Encoder): Synthetic data is generated via sampling a vector in the latent vector space shaped by encoding input sample data set

In the context of generating larger data set from smaller ones, traditional approaches rely either on drawing samples from a certain statistical models established through analysis of the existing data set, or on various data augmentation techniques, which adds some variations to the existing data, such as transformation, noise addition, etc. While these methods have been used with some degree of success in terms of increasing the amount of data set for training ML models, their utility is limited by the fact that the fidelity of the data is restricted by the approximate nature of the statistical models used, and or the ad-hoc augmentation and intervention to the existing data often results in unrealistic data.

More recently, however, a body of research work has achieved a remarkable success in synthetically generating very realistic data from example data set through the use of deep neural networks (DNN). While the details of the DNN models and architecture differ by different work, those methods largely fall into one of two categories: ones using Generative Adversarial Network (GAN) [8][9]and others using Variational Auto-Encoder (VAE)[10][11] We propose the use of VAE and its variation (e.g., Variational Recurrent Auto-Encoder [12] for generating time-series data) mainly because of its ability to generate (or "decode") new data from latent vector space learned from the real data set (See Figure 2). By sampling a vector from the latent vector space shaped by encoding the input sample data into Gaussian distribution, we will be able to systematically change the degree to which the generated data differ from the real data, an ability which cannot be readily achieved in GAN-based approaches.

These methods of data synthesis can be applied in multiple parts of our simulation, especially in generating raw data for the workloads (events and properties of entities). Examples include a set of fake aerial images generated from a specific (yet real) geographic areas, the mobility trajectory of the mobile entities generated from real trace, a set of fake images containing specific types of objects.

3.2 Generating Scenarios from Text Corpora

Another avenue for exploiting the ML techniques is generating the simulation scenario in a (semi-)automatic manner. We propose leveraging natural language processing (NLP) and information retrieval (IR) techniques to analyze text documents that describe situations and/or scenarios relevant ISR operation, and to map them onto the deployment of the events and entities. Such documents may include mission planning documents, mission reports, and other publicly available documents such as news articles and Wikipedia documents. The challenge is how to translate documents in a free- or almost-free form into a finite set of pre-defined events and entities that can be used in simulations. While the prospect of fully automating such translation still requires substantial research efforts, here we present, in no particular order, a few promising directions and practical approaches that take advantage of multiple sources of information:

- Translate a semi-structured mission plan, mission reports and general description of ISR assets (e.g., from Wikipedia), which describe the types and capabilities of assets used and the target events to be monitored, into the specification of the events and the entities.
- Address the discrepancy in the dictionary used in the free-form documents and the entities/events description, by find their mapping in vector representation models of words, phrase, and documents, such as word2vec [5].

- Mine the sequence of events contained in mission reports and news articles to create a spatio-temporal models of the events to be generated. Techniques for document summarization and keyphrase extraction, such as TextRank [6] as well as named entity recognition [7], will be instrumental for extracting only the important sentences and phrases relevant to the events of interests, out of lengthy, free-form documents.
- Once the spatio-temporal sequence of events and entities to be deployed is identified from external documents, various synthetic scenarios can be created using the generative methods, in a way similar to generating the raw data samples.

An interesting research problem in data generation is how to *compose* a realistic data out of individual pieces of atomic data (e.g., creating a realistic "scene" from images of multiple objects). In a sense, this is an inverse problem of recognizing the scene (or contexts) from data that is already composed of multiple objects. Recent work [13][14]reports on synthesizing images out of text description, though the outcome is somewhat limited to generative images of individual objects. Being able to compose realistic data will be a powerful tool for generating a data set for a particular (possibly rare) context, such as military-relevant scenarios, out of other contexts that have abundant labeled data set; for example, one can create images, sounds, and other sensory data about battlefield situations out of individual data samples of atomic elements. In practice, however, we suspect even not-so-realistic data composed of individual data pieces by heuristic methods (e.g., through re-scaling, positioning, and overlapping multiple pieces) may be useful enough for training ML models [15].

4. DATA CATALOG FOR TRAINING DATASET

Once the data sets have been generated by means of simulations, we store them into a data catalog. The data catalog allows the reuse of the generated data across several research problems and model building activities. This complementary component can be used to register, store, and query training dataset for building a specific machine learning models. The primary purposes of the data catalog are as follows:

- Provide methods for data users, in particular machine learning researchers and data analysts, to query and retrieve a set of data specific to their purpose. For this, it must be able to fulfill queries for retrieving a set of data with *specific labels* in a *specific modality* in a *specific context*. For example, one may want to build an ML model that is able to classify image data into classes of ground vehicles (labels) present in the images (modality) in an urban environment (context).
- Provide methods for data provider to register and store their datasets in an organized way such that the data can be later retrieved by data users at the sufficient level of granularity (modality, labels, contexts). For this, the data catalog provides well-defined formats/templates of meta information about the individual pieces or subsets of data, where the data labels declared in an ontological structure and the pointers to the individual pieces of data are properly associated.

Note that, although there are existing repositories for publicly available datasets for machine learning, e.g., UCI Machine Learning Repository [16], they mostly maintain a mere collection of the datasets and metadata about them, and fall short of providing the required level of granularity in data querying and retrieval.

In the context of data generation and management, our simulation-based data generation framework will primarily work as the data provider. However, it is also noteworthy that the data generation components can also benefit from the contents in data catalog as the data user, which can further synthesize new dataset for events and entities from the existing samples in the catalog, as discussed in Section 4. The following use cases describe such a constructive feedback loop between the data generator and data catalog:

- Use Case 1: Enriching the data set by further synthesizing data of a specific label
    1) Query and retrieve data for a specific modality and with a specific label.
    2) Build a VAE model with the retrieved data.
    3) Generate additional synthetic data by sampling in the latent vector space and decoding the vector into the raw data format.
    4) Registers and stores the synthesized data back in the data catalog.
- Use Case 2: Enriching the data set by composing new data for a specific context (e.g., scene of a jungle from a collection of trees and animals).

5) Determine the collection of object classes that comprise a higher-level concept (or context).
6) Query data with the specified labels.
7) Compose new data pieces from the individual labeled data.
8) Register and store the composed data in the data catalog with the new label.

5. CONCLUSION

We have presented an architecture and methods for generating realistic data set to be used as training data for building machine learning models for ISR scenarios. The scenario-driven simulation framework enables obtaining the data in a realistic setting, where the acquisition of the physical phenomena of interests, modeled as the autonomous events, by ISR sensors, modeled also as autonomous entities, is recorded via analyzing their inter-contacts, and the data quality is adjusted by the properties of the contacts, such as distance and other environmental factors. The behavior of the entities and raw data representing the physical events are generated via various methods for data synthesis, of which using generative deep-learning models is proposed as a primary approach that can systematically find a good balance between the realism and security of the raw data. We also discuss approaches to generating the simulation scenarios from corpora of text documents relevant to ISR scenarios, which has a high potential to ease the scenario generation efforts by human experts. Finally, we propose the use of a data catalog for training data set as a complementary component to the data generation utility, which can not only facilitate ML model training with data with specific contexts, but also enable incremental enrichment of the data set generation process. As our architecture allows the extension and improvement of the individual components in the data generation pipeline, we will continually explore the effectiveness of various data generation methods within our framework in the future.

6. ACKNOWLEDGEMENTS

This research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence under Agreement Number W911NF-16-3-0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copy-right notation hereon.

REFERENCES

[1] Pham, T., Cirincione, G., Swami, A., Pearson, G. and Williams, C., "Distributed analytics and information science," Proc. IEEE International Conference on Information Fusion, 245-252 (2015)
[2] Yosinski, J., Clune, J., Bengio, Y., & Lipson, H., "How transferable are features in deep neural networks?," Advances in neural information processing systems, (2014).
[3] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M. & Berg, A., "Imagenet large scale visual recognition challenge," International Journal of Computer Vision 115(3), 211-252 (2015).
[4] Liu, B., Brass, P., Dousse, O., Nain, P., & Towsley, D., "Mobility improves coverage of sensor networks," Proc. ACM international symposium on Mobile ad hoc networking and computing, 300-308 (2005).
[5] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J., "Distributed representations of words and phrases and their compositionality," Advances in neural information processing systems, 3111-3119 (2013).
[6] Mihalcea, R., & Tarau, P., "Textrank: Bringing order into text," Proc. Conference on Empirical Methods in Natural Language Processing, (2004).
[7] Nadeau, D. & Satoshi S., "A survey of named entity recognition and classification," Lingvisticae Investigationes, 30(1), 3-26 (2007).
[8] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. & Bengio, Y., "Generative adversarial nets," Advances in neural information processing systems, 2672-2680 (2014).
[9] Radford, A., Metz, L. and Chintala, S., "Unsupervised representation learning with deep convolutional generative adversarial networks," arXiv preprint arXiv:1511.06434 (2015).
[10] Kingma, D., & Welling, M., "Auto-encoding Variational Bayes," Proc. International Conference on Learning Representations, (2014)

[11] Rezende, D., Mohamed, S., & Wierstra, D., "Stochastic backpropagation and approximate inference in deep generative models," Proc. International Conference on Machine Learning, (2014)
[12] Fabius, O, & Van Amersfoort, J., "Variational recurrent auto-encoders," arXiv preprint arXiv:1412.6581, (2014).
[13] Gregor, K., Danihelka, I., Graves, A., Rezende, D. J., & Wierstra, D., "DRAW: A recurrent neural network for image generation." arXiv preprint arXiv:1502.04623 (2015).
[14] Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., & Lee, H., "Generative adversarial text to image synthesis." arXiv preprint arXiv:1605.05396, (2016).
[15] Patki, N., Wedge, R., & Veeramachaneni, K., "The synthetic data vault." Proc. IEEE International Conference on Data Science and Advanced Analytics, (2016)
[16] Dua, D. and Karra Taniskidou, E. "UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]." University of California, School of Information and Computer Science (2017).