

MSc Applied Computational Science and Engineering

Independent Research Project Plan

Name: Hameed Khandahari

Working Project Title:

Generation of multi-dimensional time-series data for fraud detection

Index terms— machine learning, deep learning, generative adversarial network (GANs), variational autoencoders (VAEs), time-series data, privacy, human-data interaction, advertising, fraud, bots, click-farming

1 Supervisor Details

1.1 External Supervisor

Name: Dr Hamed Haddadi

Email: h.haddadi@imperial.ac.uk

Website: —<https://www.imperial.ac.uk/people/h.haddadi>
—<https://haddadi.github.io/>

Position: —Senior Lecturer and Deputy Director of Research in the Dyson School of Engineering, Imperial College.

—Academic Fellow of the Data Science Institute, Imperial College

—Visiting Professor at Brave

Location: South Kensington Campus, Imperial College, SW7.

1.2 Internal Supervisor

Name: Dr Gerard Gorman

Email: g.gorman@imperial.ac.uk

Website: —<https://www.imperial.ac.uk/people/g.gorman>
—<https://haddadi.github.io/>

Position: —Reader in Computational Science, Imperial College.

Location: South Kensington Campus, Imperial College, SW7.

2 Project Summary and Motivation

2.1 Motivation

In recent years, consumers have become ever-more conscious of issues concerning their personal privacy, especially after several high profile cases such as Cambridge Analytica’s misuse of personal data collected through Facebook applications [3].

This work aims to contribute to research in the sector of electronic privacy [4] by making use of techniques prevalent in machine learning research. The research division of Brave, a privacy-focused web browser, founded by the former head of Mozilla, are particularly active in this field [1]. Brave have strong ties to Imperial College, with several academics holding various roles within the company. Amongst these are Dr Livshits, a Reader in the Mathematics department who is currently their Chief Scientist, and Dr Haddadi, the supervisor for this project, who is a Visiting Professor.

One particularly interesting aspect of Brave is its implementation of advertising. Traditionally, advertisers pay publishers, who then host advertisements which consumers interact with. This poses privacy and security issues since web advertisements can introduce trackers and malware [1][8]. These issues have led to a widespread use of third-party ad-blocking software. A study by Adobe [7] estimates that 16% of users in the United States used ad-blocking software during Q2 2015, with the estimated global loss during 2015 due to ad-blocking estimated to be \$21.8 billion.

Brave is attempting to change this ecosystem through the introduction of a system in which users are rewarded for selecting to have advertisements presented to them. This reward is in the form of BAT (Basic Attention Tokens), a cryptocurrency developed for Brave and based on Ethereum [1].

However, this model in which users are rewarded with cryptocurrencies, which can then be exchanged for conventional currencies, is open to potential exploitation by criminals who may try to ‘farm’ revenue by attempting to fool the browser into thinking that an active user is present and engaging with the advertisements presented, resulting in rewards for unauthentic ‘attention’. Similar tactics are already prevalent through the use of click-farming [5] in which arrays of smartphones are used to generate revenue through clicking on advertisements.

This project therefore ultimately aims to provide a tool which will detect and then help to prevent such behaviour by considering the sensory data of users to determine whether they are in fact genuine users. The focus will be on the browser’s smartphone applications on the iOS and Android operating systems since the sensory data of smartphones is readily available to third party applications, although work is being done on eventually restricting this due to privacy concerns [19] [23] by introducing differential privacy schemes.

2.2 Short project description

The primary aim of the project is to produce convincing synthetic time-series mobile phone sensory data in the hope of emulating the results obtained when captured by real humans. The synthetic data that is produced by the models in this project can then be used to aid with the detection of unauthentic data.

There are several approaches to achieving the first goal, the approaches used in this project are all based on deep generative models, namely Generative Adversarial Networks (GANs) and Variational AutoEncoders (VAEs). In this project, I hope to produce the synthetic data using both techniques and provide a comparison between the results obtained using the two methods and variations thereof. Much work has already been done on human activity recognition algorithms [14] [9] [24]; these can be used to analyse the outputs from the data generation models produced in this project.

Since Ian Goodfellow’s original GAN paper, there have been many publications using GANs for image generation. Some notable recent work includes Samsung’s implementation of a GAN which can produce animated headshots [25]. Further examples include CycleGAN [26], which can perform image-to-image translation and GauGAN[22], which can produce realistic images from rudimentary sketches. However there has been little work performed concerning time-series data.

In this project, I will use deep generative models, trained on an authentic multi-dimensional datasets collected by human agents, to produce synthetic time-series datasets. The work in the literature has primarily been on isolated datasets. However, for the purposes of this project, multiple datasets must be generated that concur with one other. To illustrate why an additional condition is required, let us consider accelerometer data in three directions. If this data is being collected by a human agent walking, there will be a characteristic pattern for each stride that will be present in all directions to some extent. If the three datasets are generated in isolation, there is no guarantee that these strides will align, and hence they can be easily detected as fake. Therefore, additional infrastructure needs to be included to ensure consistency between all of the data generated.

Using VAEs, specifically conditional VAEs, will provide the opportunity for interesting extensions. Let us consider a training dataset that is both large and diverse, containing significant variations in the characteristics (such as age, gender, height) of the human agents that collected them. It may be possible, in principle, to encode this information in the latent variables during the training of the VAE. If successful, it could then be possible to generate time-series data, corresponding to a human with predefined characteristics. This can be taken further by combining the sensory data with non-sensory data such as browsing activity, which is clearly dependent on the characteristics of the agent. This, however, is likely to be outside the scope of this project based on time constraints.

There are several datasets available containing time-series sensory data captured by genuine human agents carrying a smartphones. These typically contain data from the accelerometers and gyroscopes. One such dataset, **MotionSense** was collected by Mohammad Malekzadeh, a visiting researcher working on the DataBox project at Imperial College [6] and this will be the first dataset used in this project.

3 Background

In this section, a brief explanation of the most important machine learning techniques will be given and the reasons for their use will be justified. Additionally, where relevant, the implementations in the literature have been referenced. For this project plan, a basic understanding on neural networks used in machine learning has been assumed.

3.1 Generational Adversial Networks (GANs)

Introduced by Ian Goodfellow in his landmark 2014 paper [12], GANs have become an increasingly popular tool in deep learning, particularly in image generation as described previously. For the purposes of this brief explanation, it may be convinient to assume the GAN is being built for image generation to keep the nomenclature consistent with the literature.

GANs are comprised of two competing neural networks - a generator and a discriminator which compete with one another in a minimax game. A schematic is given in Figure 1.

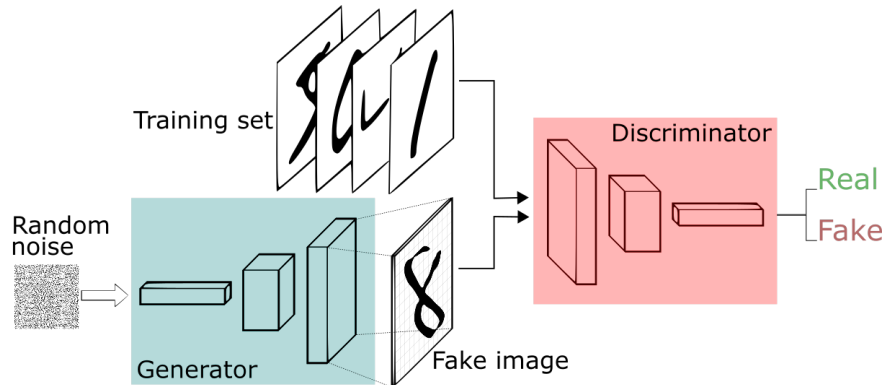


Figure 1: Schematic of a GAN. Taken from [10]

Given a high-dimension latent noise vector, which is typically sampled from a normal or uniform distribution in the literature [12] as its input, the generator network is trained to output a ‘fake’ image that is statistically close to the statistical distribution of the training set itself.

The discriminator network, however, is trained to distinguish between the results from the real training set and those produced through by the generator network. The training of each network is done in isolation using backpropagation and alternated.

The training concludes once the discriminator is unable to distinguish between samples from the training set and fake images generated using the generator network. Hence, the generator can be used in isolation without the discriminator to produce further fake images by sampling a different vector from the random distribution of latent vectors.

3.1.1 Applications

In this project, the underlying structure described above will be extended to make it more suitable for this project.

The first such extension is adding conditionality. It should be possible reproduce data corresponding to different actions, i.e. walking or jogging. This is analogous to selecting a particular digit in the MNIST dataset. This extension was originally proposed in the original Goodfellow paper [12] and was subsequently implemented in [20].

A further extension which I hope to implement is including recurrent neural networks (RNNs) in the generator and discriminator of the GAN. Recurrent neural networks capture the sequentiality of data, hence they may prove to be useful in a time-series dataset such as the ones that will be used in this project. There has been some work done using RNNs in GANs for medical applications [13] however the structure of their implementation does not ensure that multiple datasets will produce concurrent results as described in Section 2.2.

3.2 Variational AutoEncoders (VAEs)

To understand why variational autoencoders are able to generate new data, it is worth first considering autoencoders. Autoencoders use trained encoder and decoder networks to reproduce an input image [11]

The encoder takes a dataset of high dimensionality and reduces it to a latent vector of low dimensionality. The decoder then attempts to reconstruct the original dataset. By calculating the loss between the input and the reconstruction based on the input and performing backpropagation, the two neural networks can be trained to perform an effective dimensionality reduction.

Variational autoencoders work slightly differently as instead of training the decoder to produce a low-dimensional latent vector, it instead produces a distribution from which the latent vector is selected. Hence new outputs which are similar to, but not the same as, the input can be constructed [15].

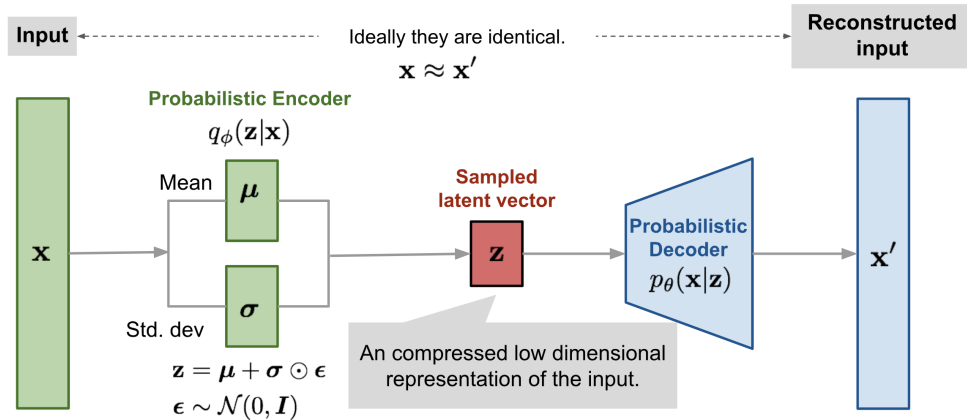


Figure 2: Schematic of a VAE. Taken from [17]

4 Progress to date

In order to familiarise myself with the process of training a GAN, I have so far attempted to provide my own implementation of a conditional GAN based existing implementations

on GitHub[2] and the paper introducing conditional GANs [20]. This network, trained on the MNIST dataset, was not intended to be a high quality, optimised network, but a first attempt at my own implementation of a GAN. A conditional VAE was also produced, also on MNIST data.







The training performed so far has been done on the Google Colab platform (<https://colab.research.google.com>). This is a free-to-use cloud-based tool which allows users to make use of an NVIDIA Tesla K80 GPU for a limit of 12 hours. Whilst this was sufficient to train the networks on the MNIST data, the actual datasets that will be used in this project will be significantly larger. Hence, I have arranged with my external supervisor to make use of Imperial College's High Performance Computing (HPC) facility for later work.

Aside from these implementations, which were intended more for my personal understanding rather than in hope of use for the actual project work, the majority of the time spent so far has been primarily on reading. I have also attended talks at the company on relevant work.

5 Milestones

Below is a summary of the key target deadlines. These may be subject to change following discussions with supervisors.

KEY:		Formal Deadline		Target Deadline		Other Important Dates
		Completed		In Progress		Behind Schedule

	03/06	•	Official project start date	
03/06 →	21/06	•	Exploring underlying theory and background	
	10/06	•	Begin writing formal plan	
	21/06	•	Literature review mostly completed	
	21/06	•	Project plan draft sent for review	
	28/06	•	Final Project Plan due	
	01/07	•	Begin formal coding	
	05/07	•	High-Level design	
	12/07	•	First GAN method implemented with time series data	
	12/07	•	VAE method implemented with time series data	
15/07 →	19/07	•	Further data collection (if required)	
15/07 →	30/08	•	Report writing	
19/07 →	15/08	•	Carrying out extensions (if on-track)	
19/07 →	15/08	•	Refinements to model for better performance	
	15/08	•	First near-complete draft of the report	
	30/08	•	Final software submission	
	30/08	•	Report submission	
	≈ 15/09	•	Oral presentation	

6 Literature Review

Some of the literature considered for the project so far can be summarised into a table, showing the main concepts within the key papers and whether they are used in the work that contributes to this project.

	Time-series	GANs	VAE	RNNs	Bots	Sensory Data
[12]		✓				
[20]		✓				
[24]	✓			✓		✓
[13]	✓	✓		✓		
[18]	✓					✓
[16]	✓	✓	✓			✓
[21]	✓	✓		✓		

References

- [1] Brave Research — Brave Browser. URL: <https://brave.com/research/>.
- [2] eriklindernoren/Keras-GAN. URL: <https://github.com/eriklindernoren/Keras-GAN>.
- [3] Facebook trust levels collapse after Cambridge Analytica data scandal - Business Insider. URL: <https://www.businessinsider.com/facebook-trust-collapses-after-cambridge-analytica-data-scandal-2018-4?r=US&IR=T>.
- [4] IEEE Symposium on Security and Privacy 2019. URL: <https://www.ieee-security.org/TC/SP2019/>.
- [5] Inside Of A Chinese Click Farm (10,000+ phones) - YouTube. URL: <https://www.youtube.com/watch?v=NXvzhYn1TU0>.
- [6] motion-sense directory. URL: <https://github.com/mmalekzadeh/motion-sense/tree/master/codes>.
- [7] The cost of ad blocking PageFair and Adobe 2015 Ad Blocking Report. Technical report. URL: https://downloads.pagefair.com/wp-content/uploads/2016/05/2015_report-the_cost_of_ad_blocking.pdf.
- [8] Sajjad Arshad, Amin Kharraz, and William Robertson. Identifying Extension-based Ad Injection via Fine-grained Web Content Provenance. Technical report. URL: <https://arxiv.org/pdf/1811.00919.pdf>.
- [9] Akram Bayat, Marc Pomplun, and Duc A. Tran. A Study on Human Activity Recognition Using Accelerometer Data from Smartphones. *Procedia Computer Science*, 34:450–457, 2014. URL: <https://linkinghub.elsevier.com/retrieve/pii/S1877050914008643>, doi:10.1016/j.procs.2014.07.009.

- [10] Antonia Creswell and Anil Anthony Bharath. Adversarial Training for Sketch Retrieval. pages 798–809. 2016. URL: http://link.springer.com/10.1007/978-3-319-46604-0_55, doi:10.1007/978-3-319-46604-0{_}55.
- [11] Carl Doersch. Tutorial on Variational Autoencoders. Technical report, 2016. URL: <https://arxiv.org/pdf/1606.05908.pdf>.
- [12] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. Technical report. URL: <http://www.github.com/goodfeli/adversarial>.
- [13] Stephanie L Hyland, Eth Zurich, Cristóbal Esteban, and Gunnar Rätsch ETH Zurich. REAL-VALUED (MEDICAL) TIME SERIES GENERATION WITH RECURRENT CONDITIONAL GANS. Technical report. URL: <https://arxiv.org/pdf/1706.02633.pdf>.
- [14] Andrey Ignatov. Real-time human activity recognition from accelerometer data using Convolutional Neural Networks. *Applied Soft Computing*, 62:915–922, 1 2018. URL: <https://linkinghub.elsevier.com/retrieve/pii/S1568494617305665>, doi:10.1016/j.asoc.2017.09.027.
- [15] Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. Technical report. URL: <https://arxiv.org/pdf/1312.6114.pdf>.
- [16] Nikolay Laptev. AnoGen: Deep Anomaly Generator. Technical report. URL: <https://research.fb.com/wp-content/uploads/2018/11/AnoGen-Deep-Anomaly-Generator.pdf>?
- [17] Lilian Weng. From Autoencoder to Beta-VAE, 2018. URL: <https://lilianweng.github.io/lil-log/2018/08/12/from-autoencoder-to-beta-vae.html>.
- [18] Mohammad Malekzadeh, Richard G Clegg, Andrea Cavallaro, and Hamed Haddadi. Mobile Sensor Data Anonymization. 10(19), 2019. URL: <https://doi.org/10.1145/3302505.3310068>, doi:10.1145/3302505.3310068.
- [19] Mohammad Malekzadeh, Richard G. Clegg, and Hamed Haddadi. Replacement AutoEncoder: A Privacy-Preserving Algorithm for Sensory Data Analysis. In *2018 IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation (IoTDI)*, pages 165–176. IEEE, 4 2018. URL: <https://ieeexplore.ieee.org/document/8366986/>, doi:10.1109/IoTDI.2018.00025.
- [20] Mehdi Mirza and Simon Osindero. Conditional Generative Adversarial Nets. Technical report. URL: <https://arxiv.org/pdf/1411.1784.pdf>.
- [21] Olof Mogren. C-RNN-GAN: Continuous recurrent neural networks with adversarial training. Technical report. URL: <https://github.com/olofmogren/c-rnn-gan>.

- [22] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic Image Synthesis with Spatially-Adaptive Normalization. Technical report. URL: <https://arxiv.org/pdf/1903.07291.pdf>.
- [23] Sandra Servia-Rodriguez, Liang Wang, Jianxin R. Zhao, Richard Mortier, and Hamed Haddadi. Privacy-Preserving Personal Model Training. In *2018 IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation (IoTDI)*, pages 153–164. IEEE, 4 2018. URL: <https://ieeexplore.ieee.org/document/8366985/>, doi:10.1109/IoTDI.2018.00024.
- [24] Jindong Wang, Yiqiang Chen, Shuji Hao, Xiaohui Peng, and Lisha Hu. Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters*, 119:3–11, 3 2019. URL: <https://linkinghub.elsevier.com/retrieve/pii/S016786551830045X>, doi:10.1016/j.patrec.2018.02.010.
- [25] Egor Zakharov, Aliaksandra Shysheya, Egor Burkov, and Victor Lempitsky. Few-Shot Adversarial Learning of Realistic Neural Talking Head Models. Technical report. URL: <https://arxiv.org/pdf/1905.08233.pdf>.
- [26] Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A Efros, and Berkeley Ai Research. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks Monet Photos. Technical report. URL: <https://arxiv.org/pdf/1703.10593.pdf>.