Mobile Networks and Applications (2020) 25:977–985
https://doi.org/10.1007/s11036-019-01233-7

Intelligent and Good Machines? The Role of Domain and Context Codification

Giovanni Delnevo[1] · Marco Roccetti[1] · Silvia Mirri[1]

Published online: 26 February 2019

Abstract

There is a core problem with the modern Artificial Intelligence (AI) technologies, based on the current new wave of Artificial Neural Networks (ANNs). Whether they have been used in healthcare or for exploring Mars, we, the programmers who build them, do not know well why they make some decisions over others. Many are putting into question, hence, this aura of AI objectivity and infallibility; on our side, we, instead, identify a key issue around the problem of AI errors and bias into an insufficient human ability to determine the limits of the context, where the ANNs will have to operate. In fact, while it is of great amplitude the range of what the rational side of the human mind can master, machine intelligence has limited capacity to learn in completely unknown scenarios. Simply, an inaccurate or incomplete codification of the context may result into AI failures. We present here a simple cognification ANN-based case study, in an underwater scenario, where the difficulty of identifying and then codifying all the relevant contextual features has led to a situation of partial failure. This paper reports on our reflections, and subsequent technical actions taken to recover from this situation.

Keywords Artificial intelligence · Interactive machine learning · Transductive · Transfer learning · Neural networks · Context formalization

1 Introduction

The debate on the implications of the use of AI for people and, in general, for the society is still open [1–9]. It becomes even more controversial in contexts where automatized decisions on crucial factors can have a deadly impact, like healing a patient or enforcing law [10–15]. Many conflicting positions are emerging. On one side, Mark Zuckerberg considers this AI as an unprecedented means to eradicate hitherto indestructible problems that have afflicted humanity for a long time; on the other side, Elon Musk, Bill Gates and Stephen Hawking have emphasized the risk that humanity is running, with a high probability of becoming second-class citizens. After all these premises, our modest vision is that of a people-centric AI, where humans help improve machine learning algorithms.

The problem is that while human beings are optimized for learning unlimited patterns, and then selecting the most appropriate one to deal with whatever situation they encounter, this is different for machines. In fact, often AI errors and bias occur due to the insufficient and inaccurate capacity of: i) identifying the domain range, and ii) understanding and codifying the correspondent context where the ANNs have to operate [16–20]. Disregarding these factors means ignoring that grey area where often human intelligence gives its best. Underestimating the role of the context is equivalent to misunderstanding the fact that often machines should be designed not to emulate humans, rather to collaborate with them. In simple words, and translating this into the machine learning world, ignoring the context can lead to a wrong training activity of ANNs and, hence, to dangerous results.

Unfortunately, we have a typical big problem in designing and training a simple and efficient ANN, able to deal with large datasets containing hundreds of thousands of pieces of information, while managing an intelligible and accurate formalization of the context where it has to operate. What often is needed, indeed, is to train machines to work with more *details* that match this specific *context*.

✉ Giovanni Delnevo
giovanni.delnevo2@unibo.it

Marco Roccetti
marco.roccett@unibo.it

Silvia Mirri
silvia.mirri@unibo.it

[1] Department of Computer Science and Engineering, University of Bologna, Bologna, Italy

In this paper, we present our experience in the cognification of an underwater scenario where ANNs were trained to find optimal routes, like for example in the case when an underwater fiber optic cable needs to be installed [21, 22].

Obviously, we are well aware that pathfinding problems are often tackled and modeled through the use of traditional optimization techniques. Yet, we are not here to question those traditional approaches, as their efficacy has been often demonstrated in several contexts. However, those approaches typically leverage on either basic or complex decision logics, i.e. a set of local/general rules to be satisfied within a specific context, in the presence of all the data that need to fully describe it. The situation we are encountering here (underwater scenario) is, instead, at the opposite end of the spectrum. Even after a vessel inspects the seabed, using appropriate instrumentations, it just returns a subset of all that relevant information (e.g. soil type) that is then used by humans (geologists) to establish an optimal underwater route.

All this said, our intent here, is to understand to which extent we are able to let a neural network learn from incomplete data and patterns, still aiming at achieving an underwater route that satisfies human standards. The unavailability of all the data that could feed a traditional optimization problem motivates our ANN-based choice. Nonetheless, this learning process is a complex task, where an initial and incomplete knowledge has to be reinforced through additional ANN-based techniques, like those proposed in this paper. There is plenty of similar approaches in literature. Just to cite a few: [23] in the field of computer gaming; [24] in the field of robotic movements; and finally [25] in the context of image interpretation. With our running example, we report on our reflections, and consequent technical actions, we have made to train a neural network, with the final aim to adjust it to a domain whose context may be highly unstable and variable in nature and in time.

The reminder of the paper is structured as follows. Section 2 illustrates an initial ANN training scenario where many crucial factors of the context were ignored with consequent negative results. In Section 3, instead, we show how the introduction of those factors ameliorates the final results. Section 4 presents the results we have obtained in a different domain with the same ANN. Finally, Section 5 concludes the paper.

2 The role of the context

We trained an ANN with real data coming from an underwater fiber optic cable installation case. We have utilized a feed forward ANN with 50 neurons in the hidden layer, of the type described in [26], motivated by its renown flexibility and ability to adapt to different training conditions. The situation, in the reality, is as follows. Before the crew of a vessel install a cable, an a-priori in-depth analysis of the seabed takes place.

Table 1 African seabed: soil types

1	coarse sediment
2	fine sediment
3	rock
4	subcropping rock
5	seagrass
6	gas charged sediment
7	depressions
8	slump
9	scars
10	pipe

The vessel inspects the seabed, obtaining a grid of sampled points of which the depth and the soil typology are the main information. Afterwards, using these data, a team of geologists define the optimal route. This route is the final output of the analysis process and it is provided to the crew of the vessel that will then install the cable.

In this context, we trained an ANN to find an optimal route of a given African seabed. In particular, here, the domain can be seen as defined by the type of the soil of this specific portion of this African seabed. Specifically, Table 1 portrays the ten different types of soil. Soil typology is important for evident motivations. For example, if at a given point we have a rock, no cable can be buried, due to the obstruction represented by the rock. The presence of gas is another hazardous factor to be avoided, while the sand and fine sediments are optimal soil typology, since they do not cause any risk to both the cable and the vessel crew.

Essentially, our ANN has been trained to learn a path of subsequent *good points*, based on optimal routes provided by geologists, who decided the next direction to take, depending on the soil type of the surrounding points. Basically, our ANN learns how to move on a bidimensional grid, like that in Table 2. In particular, given the current point (x,y), at the center of the grid, our ANN has a freedom of choice in terms of movements which has no limits. In essence, it can go straight up to reach point 6, left-up to reach point 5, right-up to reach point 7, and so on to reach each and any different point within the grid. No other contextual information was initially formalized and passed to the ANN to be learnt.

After the training phase, we then run our ANN and contrasted the obtained results against those provided by the team of geologists who were our advisors on this case. After several experiments, we found evidence that

Table 2 Possible movements

5	6	7
4	x,y	0
3	2	1

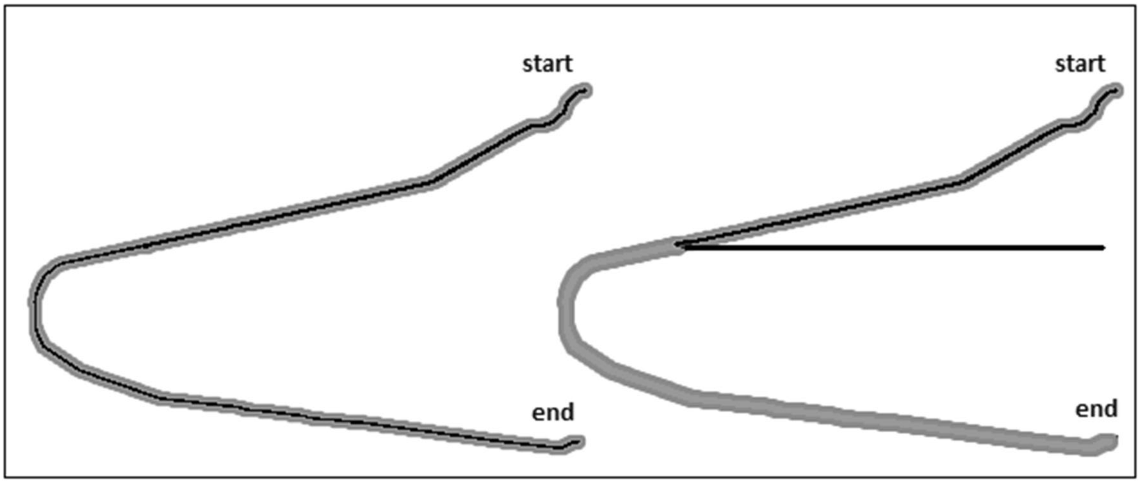


Fig. 1 Route comparison. Geologists (leftmost) vs. ANN (rightmost); route (black), corridor (grey)

the routes computed by the ANN deviated from the trajectories suggested by geologists, going outside of the inspected seabed corridor, without even reaching the destination, as emphasized in the example of Figure 1.

With the aim of ameliorating this situation, without altering the training process, we simply added some further rules to be executed after the ANN had run. In essence, we developed a backtracking mechanism. If the ANN goes out of the inspected seabed corridor, like in the example above, we remove the point from the route and let the ANN re-compute an alternative path, trying to identify an admissible route within the corridor.

With this a-posteriori modification, our ANN has become always able to suggest a plausible route, from the starting point to the destination, as shown in the example of Figure 2. Nonetheless, this good result has come with the drawback that these computed routes can be often much longer than those proposed by geologists. This phenomenon is evident in Figure 2, where a careful analysis of the rightmost route highlights a highly frequence serrated line. That particular serrated line amounts to a collection of subsequent segments characterized by slight variations along the direction. In conclusion, this implies longer routes, as portrayed, for example, in Table 3, third column, where we have reported some instances of the route length ratio (ANN / geologists). Basically, our idea to use external rules and logic (e.g. the backtracking algorithm) to correct, in some sense, a pure machine learning mechanism does not work well. If we do not use the backtracking algorithm the computed route often gets out of the corridor, nonetheless if we use it, we obtain correct, yet much longer, pathways towards the destination. At the end, we have to admit that we have to change our approach to this problem. We will discuss this in the next Section.

3 Sharpen the focus

The discrepancy between the routes computed by our ANNs and those provided by the geologists suggests us that the

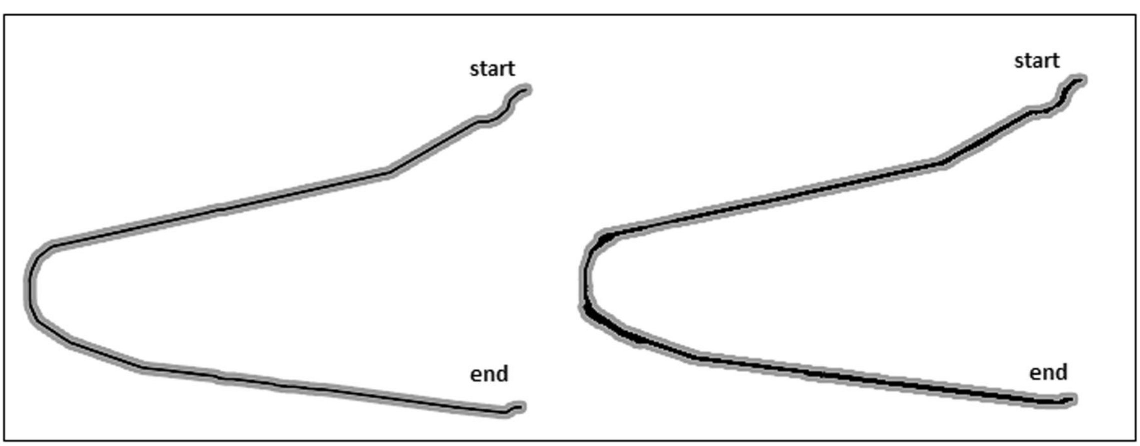


Fig. 2 Route comparison II. Geologists (leftmost) vs. eural network (rightmost); route (black), corridor (grey)

Table 3 Route length ratio (ANN / geologists)

Instance	Out	Ratio
1	No	4.98
2	No	4.53
3	No	5.71

codification of the context, so far, did not contain all the factors relevant to the case (nature and characteristics of the cable, for example). Hence, our decision to sharpen the focus on the context, looking for relevant factors, previously not considered. We identified the following three ones:

1. The direction along which a cable is pulled is relevant (e.g., pulling a cable from top to bottom or vice versa has different implications).
2. Out of all the possible angular movements (0, 45, 90, 135, and 180 degrees) only those that do not implicate a cable inversion should be allowable, depending on the orientation of the corridor.
3. The cable cannot get out of the inspected seabed corridor.

3.1 Learning process

Based on the considerations above, we tried to change our model of the context, including the factors previously mentioned. First of all, we focused on the direction along which the cable is pulled. The problem with our initial model emerges from the following example. Consider the situation in Table 2. A direction from bottom to top. Rocks (dangerous soil) in point 5 and 6, sand (good soil) in point 7. The right decision to make is moving from the center of the grid to point 7. Now, consider a specular situation when only the direction of the movement is changed (from top to bottom), sand in point 1, rocks in point 2 and 3. The right decision, in this case, is to move from the center of the grid to point 1.

This explains well that an explicit codification of this contextual information is needed to allow our ANN to be trained correctly. This new domain codification is reported in Table 4 where all the possible cases are shown along with the correspondent direction to be followed. In essence, we have made explicit the relationship between the direction to take (i.e., 0, 1, 2, 3, 4, 5, 6, and 7) and the orientation of the corridor (the arrow).

Go to problem 2 now. We cannot allow both those movements that imply that our ANN goes back and forth (a 180-degrees inversion) or those that determine a sudden change in the direction (135-degrees). In fact, this should be a nonsensical decision. This can be simply codified, and then passed to our ANN to be learnt, by making not admissible those precise movements that are shown in Table 5. Not only, the cable cannot make a 90-degrees bend, otherwise, it breaks. This can be simply formalized, and then passed to our ANN to be learnt, by making not admissible those precise movements that make a 90-degrees bend.

Table 5 Possible movements revised

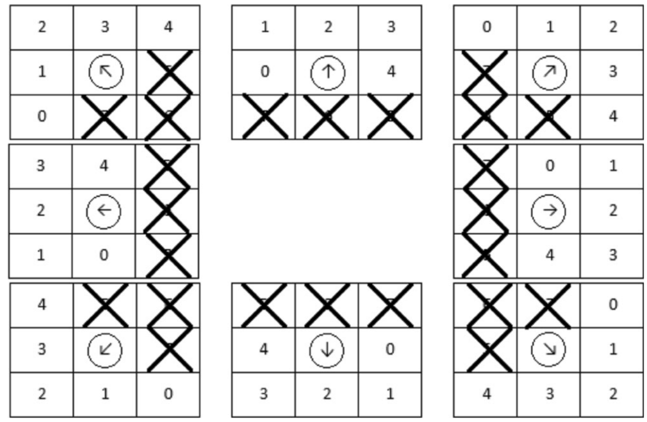


Table 4 Possible movements

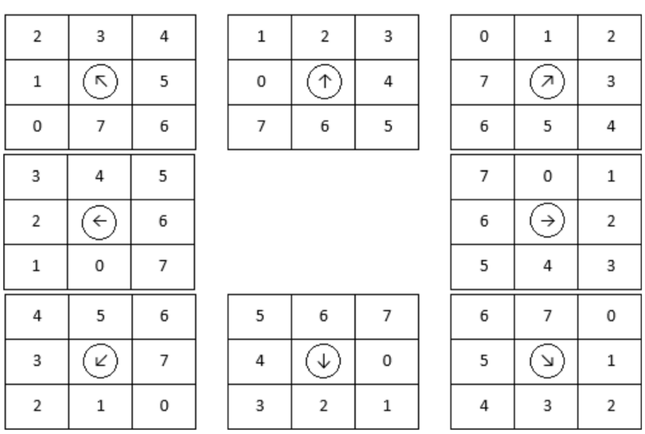
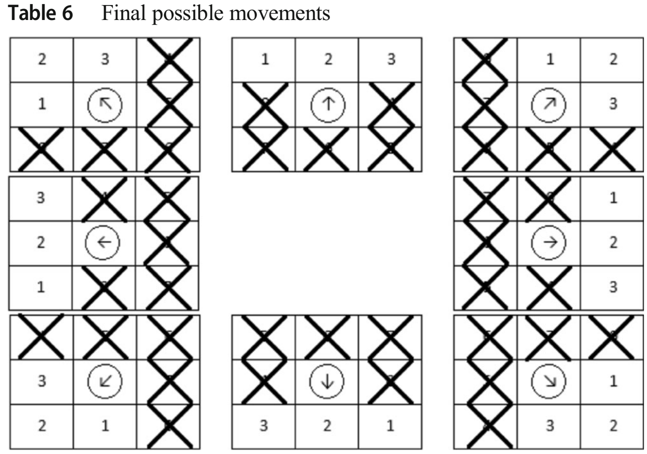


Table 6 Final possible movements



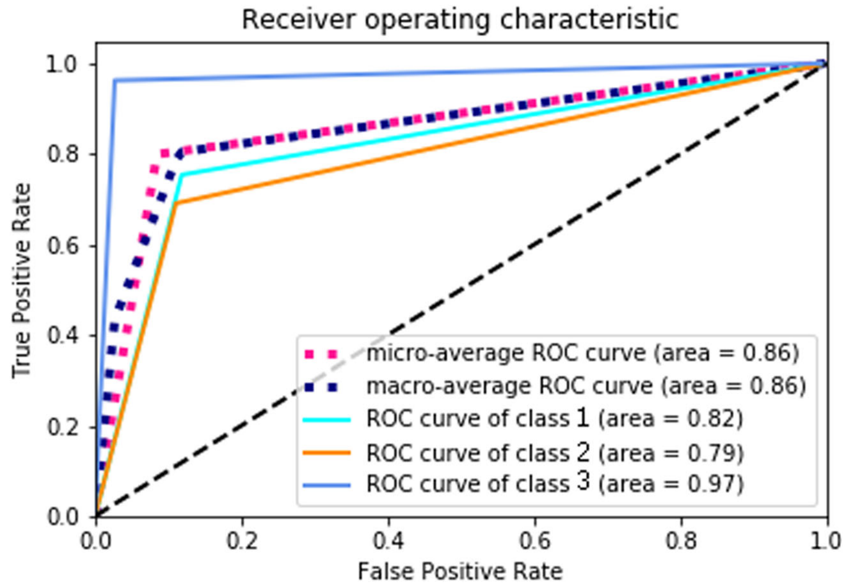


Fig. 3 Movements: AUC values

All these contextual factors, concerning the cable, can be hence codified as per the new and final revised Table 6. After the formalization of points 1 and 2, we trained again our ANN and run it to check its behavior. Consider now the results we obtained. Figure 3 shows how many times our ANN took the (right) decision to go to points 1, 2 and 3 (the only admissible ones), while ignoring all the others. The correct behavior of our ANN is confirmed by the value of 0.82, 0.79, 0.97, respectively associated to the movements towards points 1, 2 and 3, of the area under the Receiver Operating Characteristic (ROC) curves. Also the average value of those figures of merit (0.86) confirmed the efficacy of our new context formalization.

Figure 4, nonetheless, reveals that we still have a problem. In fact, even if the formalization of factors 1 and 2 have made our ANN tolerant to the typical cable problems that initially were ignored, it still disregards the important fact that our ANN cannot choose a route that exits the inspected seabed corridor, for no reason. This is evident from an analysis of Figure 4, where one example of the routes computed by our newly retrained ANN is shown.

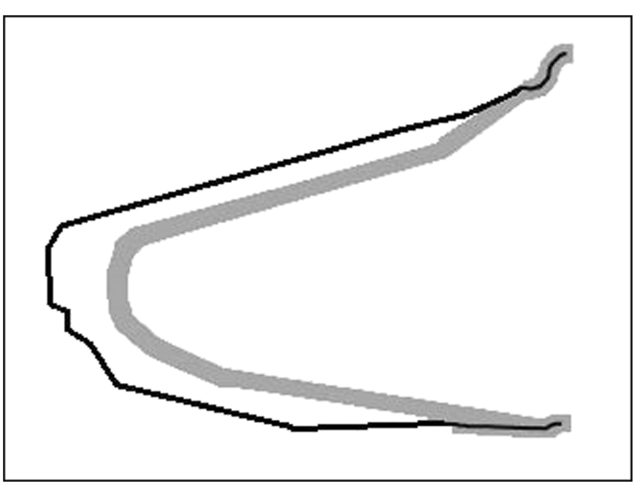


Fig. 4 Route (black) outside the corridor (grey)

3.2 Learning through failures

A major and new problem here does emerge. Human knowledge is limited and humans are afraid of all what is out of domain of their knowledge. This is exactly the case. A vessel returns to us information about the soil type of a given seabed corridor. Geologists consider as acceptable only those routes that fall within that given corridor. For reasons we are not titled to understand, our ANN, instead, often tends to exit that corridor, as Figure 4 has confirmed. We need, hence, a new method to pass to our ANN this kind of contextual information: "Please don't get out of the corridor". To this aim, following the well-known interactive machine learning paradigm [27], we designed and implemented a procedure that lets the ANN learn this information, based on a typical *learning through failures* approach [28–30].

Our technique goes as follows. We take our trained ANN and we run it. When the network predicts a point to follow, we

Table 7 Number of errors

	Step 1	Step 2	Step 3
ANN 1	20	14	0
ANN 2	285	0	–
ANN 3	10	0	–
ANN 4	30	8	0
ANN 5	1	0	–
ANN 6	8	0	–

Table 8 ANNs: route length

Instance	Out	Ratio
1	No	0.998
2	No	1.010
3	No	1.021
4	No	1.000
5	No	1.017
6	No	1.019
7	No	1.013
8	No	1.019
9	No	1.025
10	No	1.018
11	No	1.005
12	No	1.019

Table 9 New seabed, new domain

1	coarse sediment	12	escarpment
2	fine sediment	13	hardened sea floor
3	rock	14	linear sonar contact
4	subcropping rock	15	magneto
5	gas charged sediment	16	pock
6	slumps	17	sonar contact
7	scars	18	sond
8	pipe	19	stiffclay
9	cable	20	wreck
10	conefacies	21	vcs
11	debris		

control that the chosen point is within the corridor. If not so, we store this information in a list (call them error and list of errors, respectively), yet we eliminate it from the route and ask the ANN to re-compute the next point to follow until an admissible one, within the corridor, is chosen. At the end of this phase, we have an admissible route and a list of errors on which we re-train our ANN. In essence, the implicit knowledge represented by all the deviations from the correct route (i.e., the errors) become a fundamental subject to be learnt. We then repeat this process "train-run-error-learn from errors" iteratively until the point no route point is computed by our ANN that is out of the given corridor.

To prove the validity of our approach, we have replicated this process six times with six different ANNs (six different training configurations). Table 7 demonstrates that this process has converged for all the six cases we have examined, in the sense that all the six considered ANNs make no errors after a three-step long training process.

To check whether our method can be generalized, we have taken yet another new ANN and tried to train it with all the information (both positive and negative) that have stemmed from the previously described process of Table 7, where exactly 376 errors were made in total, out of 3667 route points. (Not) Surprisingly, this seventh ANN always remains within the seabed corridor, never making any error.

To be sure that the case of our seventh ANN was not an isolated situation, we repeated the process with other eleven different ANNs (eleven different training configurations), each one learning the information summarized in Table 7. After this further experiment, we can confirm that no one of them got out of the corridor as shown in Table 8 (second column). Table 8 portrays another interesting information. All our new ANNs, retrained based on the process described above, have a length which is comparable to the one provided by the geologists, as shown by the correspondent route length ratio in the third column.

We also conducted a kind of sensitivity analysis, to understand if a smaller amount of errors could be sufficient (less

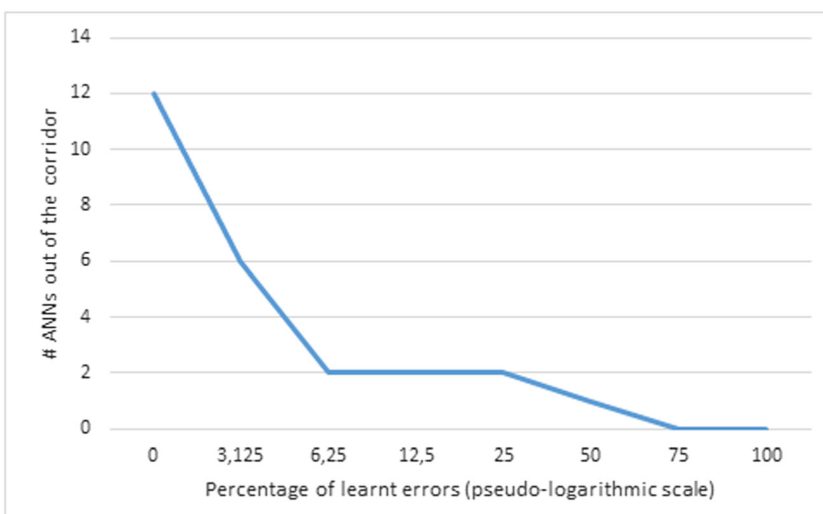


Fig. 5 ANNs: remaining within the corridor

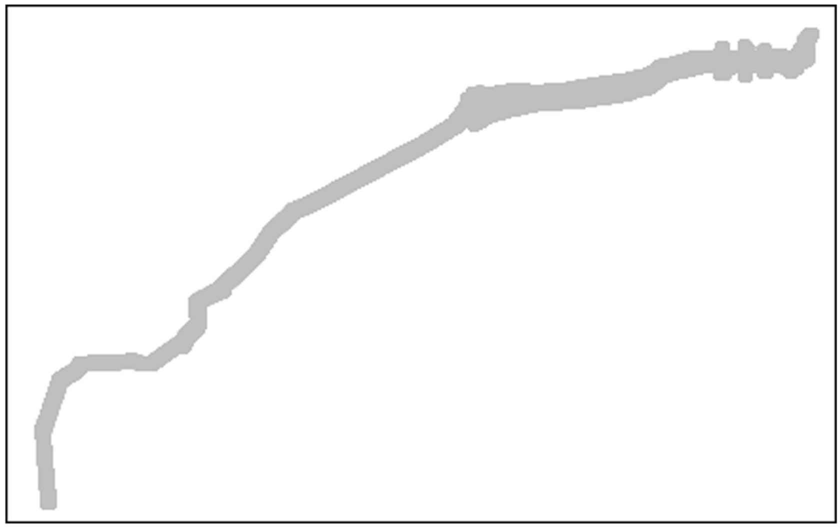


Fig. 6 New seabed corridor

than 376) to let our ANNs learn how not to get out of the corridor. This analysis has produced the graph of Figure 5, where it is shown that all the twelve ANNs are able to remain within the corridor, just showing to them a 75% of the total amount of errors, randomly chosen on the basis of a uniform distribution (precisely 282 errors out of 376).

4 Domain change and readjustment

So far, the complex training activity we have developed to let our ANNs learn the African seabed seems to work well. A natural question, nonetheless, is: what happens if the domain changes? Should the training process, we have defined and implemented, be restarted again from scratch?

To respond to these latter research questions, we took into consideration a completely new seabed whose: i) soil typologies are shown in Table 9 (with terms from the geological jargon) and ii) the general shape is portrayed in Figure 6. This examined seabed is evidently different from the African one (we are not allowed to reveal the exact geographical position). Differences are not only those visible in Table 9 and Figure 6, yet also amount to the different route length that is almost 333km (contrasted against the length of 216km of the African seabed).

At this stage, we have taken one of the best neural networks we had trained before, namely the number 1 of Table 8, and asked it to find an optimal route for this seabed corridor. Not surprisingly, the results are not so good, with 436 different points where the computed route gets out of the new corridor (new errors to be learnt). Our lack of surprise around these negative results is motivated by the fact that this new portion of seabed include thirteen new different soil typologies that no trained network, among ours, had ever seen before.

Table 10 New seabed: number of errors and length ratios

Step	Errors	Ratio
1	436	1,09
2	31	1,08
3	0	1,08

At this point, we have two possible alternatives. The first one amounts to train a brand-new network with the new route, starting from the beginning of this long process. For sure, this *traditional* approach would lead to a positive result, yet implicating that our efforts for a context codification, so far conducted, would have been lost, thus resulting in huge waste of time and knowledge.

Alternatively, a choice following the specific line of our research could be that of retraining one of our old networks to learn just the failures suffered during the first exploration of this new seabed. Resorting to the literature, this process falls within the category of *transductive transfer learning*: the task being the same, yet within a different domain [31].

Hence, we subjected our network to a process of learning from those 436 failures. This process went into convergence after three steps, as portrayed in Table 10, in which both the number of errors before convergence, and the correspondent route length ratios are shown. One of the routes generated at the third step is portrayed in Figure 7.

A final remark is due concerning the computational time needed to carry out the process we have presented in this paper. We have not stressed before this crucial factor, simply for the fact that all the neural network training activities have taken, at most, order of some minutes to complete. Instead, more time consuming have been all the ANN execution activities (upon completion of the training activity). This is due to motivations that go beyond the typical neural network execution issues. This waste of time is simply due to the specific nature of the geographical problem, where complex grids of points have to be assembled in real time, while the neural network is in execution (orders of hours).

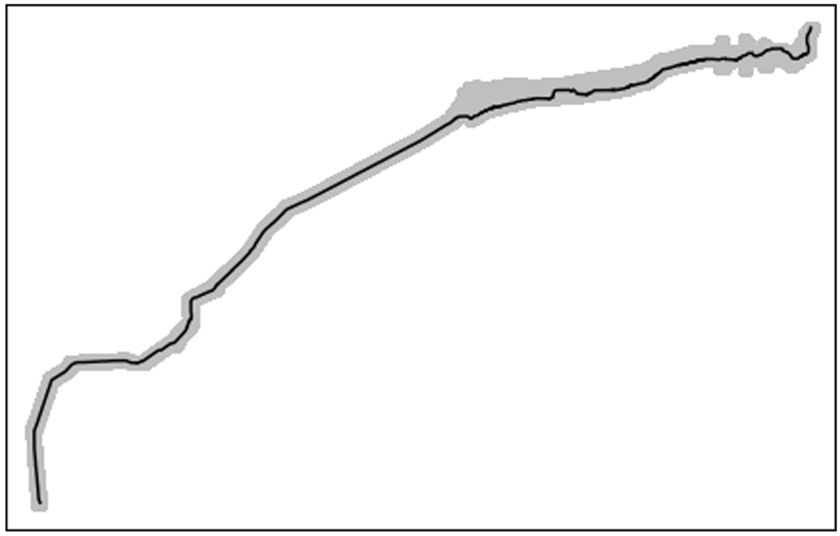


Fig. 7 New seabed, a correct route

To conclude our discussion, we have demonstrated that a nuanced consideration of a complex context, with subsequent technical actions, can be effective for the training process of a neural network, even in the presence of highly variable domains.

5 Conclusion

There has been much commentary about the negative potential for ANNs to have a detrimental impact to humans and to the society. Our vision is that a comprehension and formalization of the domain/context where the ANNs have to operate can smooth out this problem. Obviously, a complex task is devising an efficient and speedy method to train an ANN, while managing an intelligible and accurate formalization of the context of interest. We have presented a case study where the comprehension and codification of contextual information have improved the behaviour of an ANN, making it more *human*, in some sense.

Acknowledgements We are grateful to the company that has provided us with the experimental seabeds. Topologies and company name are kept secret for the sake of privacy and security. Our thanks also go to Maurizio Sobrero, Andrea Lipparini, Lorenzo Donatiello and Gustavo Marfia (University of Bologna) for many precious suggestions on a previous version of this manuscript.

References

1. Lemley J, Bazrafkan S, Corcoran P (2017) Deep learning for consumer devices and services: pushing the limits for machine learning, artificial intelligence, and computer vision. IEEE Consum Electron Mag 6.2:48–56
2. Wu Y, Schuster M, Chen Z, Le QV, Norouzi M, Macherey W, Krikun M, Cao Y, Gao Q, Macherey K, Klingner J, Shah A, Johnson M, Liu X, Kaiser L, Gouws S, Kato Y, Kudo T, Kazawa H, Stevens K, Kurian G, Patil N, Wang W, Young C, Smith J, Riesa J, Rudnick A, Vinyals O, Corrado G, Hughes M, Dean J (2016) Google's neural machine translation system: bridging the gap between human and machine translation. CoRR abs/1609.08144 (2016). arXiv:1609.08144 http://arxiv.org/abs/1609.08144
3. Van Den Oord A, Dieleman S, Zen H, Simonyi K, Vinyals O, Graves A, Kalchbrenner N, Senior AW, Kavukcuoglu K (2016) WaveNet: A generative model for raw audio. In SSW 125
4. Son Chung J, Zisserman A (2018) Learning to lip read words by watching videos. Comput Vis Image Underst
5. S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman. 2017. Synthesizing Obama: learning lip sync from audio. ACM Trans. Graph. 36, 4, Article 95 (July 2017), 13 pages. doi:https://doi.org/10.1145/3072959.3073640
6. Chen F, Li B, Dong R, Zhao P (2018) High-performance OCR on packing boxes in industry based on deep learning. Pacific Rim International Conference on Artificial Intelligence Springer, 1018–1030
7. Silver D, Huang A, Maddison CJ, Guez A, Sifre L, Van Den Driessche G, Schrittwieser J, Antonoglou I, Panneershelvam V, Lanctot M et al (2016) Mastering the game of go with deep neural networks and tree search. Nature 529 7587(2016):484
8. Moravˇc'ık M, Schmid M, Burch N, Lis'y V, Morrill D, Bard N et al (2017) Deepstack: expert-level artificial intelligence in heads-up no-limit poker. Science 356(6337):508–513
9. AI O, (2018) The open ai dota 2 bot. [Accessed 20-August-2018]. [Online]. Available: https://blog.openai.com/more-on-dota-2/
10. Ferretti S, Mirri S, Prandi C, Salomoni P (2016) Automatic web content personalization through reinforcement learning. J Syst Softw 121:157–169. https://doi.org/10.1016/j.jss.2016.02.008
11. Roccetti M, Salomoni P, Prandi C, Marfia G, Montagnani M, Gningaye L (2016) Understanding Crohn's disease patients reaction to infliximab from facebook: A medical perspective. Proceedings of the 2016 IEEE/ACM international conference on advances in social networks analysis and mining, ASONAM 2016. 1007 – 1010. doi:https://doi.org/10.1109/ASONAM.2016.7752364
12. Roccetti M, Prandi C, Salomoni P, Marfia G (2016) Unleashing the true potential of social networks: confirming infliximab medical trials through Facebook posts. Netw Modeling Anal Health Inform Bioinform 5:1. https://doi.org/10.1007/s13721-016-0122-9

13. Silvia Mirri, Catia Prandi, Marco Roccetti, and Paola Salomoni. (2017) Handmade narrations: handling digital narrations on food and gastronomic culture. ACM J Comput Cult Herit. doi:https://doi.org/10.1145/3097569
14. Ferretti S, Mirri S, Prandi C, Salomoni P (2017) On personalizing web content through reinforcement learning. Univ Access Inf Soc 2017. https://doi.org/10.1007/s10209-016-0463-2
15. Melis A, Mirri S, Prandi C, Prandini M, Salomoni P, Callegati F (2018) Integrating personalized and accessible itineraries in MaaS ecosystems through microservices. Mobile Networks and Applications 2018. https://doi.org/10.1007/s11036-017-0831-z
16. Caruana R, Lou Y, Gehrke J, Koch P, Sturm M, Elhadad N (2015) Intelligible models for healthcare: predicting pneumonia risk and hospital 30-day readmission. Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining. ACM: 1721–1730
17. Melis A, Mirri S, Prandi C, Prandini M, Salomoni P, Callegati F (2016) Crowd sensing for smart mobility through a service-oriented architecture. IEEE 2nd Int Smart Cities Conf: Improving Citizens Qual Life, ISC2 2016. https://doi.org/10.1109/ISC2.2016.7580861
18. Mirri S, Prandi C, Salomoni P, Monti L (2016) Social location awareness: A prototype of altruistic IoT. 2016 8th IFIP International Conference on New Technologies, Mobility and Security, NTMS 2016. https://doi.org/10.1109/NTMS.2016.7792477
19. Zou J, Schiebinger L (2018) AI can be sexist and racist – it's time to make it fair. Nature 559:324–326
20. Buolamwini J, Gebru T (2018) Gender shades: intersectional accuracy disparities in commercial gender classification. Conf Fairness Account Transparency 77–91
21. Delnevo G, Lipparini A, Roccetti M, Sobrero M (2018) Human intelligence, ambidexterity and ai: reflections on a data cognification case study in an underwater scenario. 2nd annual science fiction prototyping conference 2018: designing your future with science fiction. SciFi-It 2018(2018):50–55
22. Delnevo G, Roccetti M, Mirri S (2018) Intelligent machines for good?: more focus on the context. Goodtechs '18 international conference on smart objects and Technologies for Social Good Bologna, Italy
23. Graham R, McCabe H, Sheridan S (2004) Neural networks for real-time pathfinding in computer games. ITB J 5(1):21
24. Kindermann T, Cruse H, Dautenhahn K (1996) A fast, three-layer neural network for path finding. Netw Comput Neural Syst 7(2): 423–436
25. Kim H, Son H, Roska T, Chua LO (2002) Optimal path finding with space- and time-variant metric weights via multi-layer CNN. Int J Circuit Theory Applic 30(2–3):247–270
26. Benardos PG, Vosniakos G-C (2007) Optimizing feedforward artificial neural network architecture. Eng Appl Artif Intell 20.3:365–382
27. Fails JA, Olsen DR Jr. (2003) Interactive machine learning. Proc 8th Int Conf Intell User Interf. ACM
28. Arino A, De La Torre J (1998) Learning from failure: towards an evolutionary model of collaborative ventures. Organ Sci 9(3):306–325
29. Coelho PRP, McClure JE (2005) Learning from failure. Am J Business 20(1):1
30. Shepherd DA (2003) Learning from business failure: propositions of grief recovery for the self-employed. Acad Manag Rev 28(2): 318–328
31. Pan SJ, Yang O (2010) A survey on transfer learning. IEEE Trans Knowl Data Eng 22.10:1345–1359