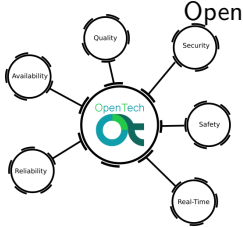
Herds of Dumb Models

A New Approach Towards Reliable and Safe AI

Nicholas Mc Guire, Imanol Allende, Carles Hernández

OpenTech EDV Research GmbH, Universitat Politécnica de Valéncia



September 20, 2025

Disclaimer

- This is low TRL but results are, we think, exciting never the less
- Current state proof-of-concept with respect to implementation
- In discussion with members of certification bodies
- Target domain: highly autonomous decisions where human intervention is unreasonable or impossible (isolated systems) e.g. deep-space missions, deep-see missions, critical decisions in autonomous systems.
- Operational Design Domain: open environments where we must expect inputs that are totally different than any training or validation data
- There is a relatively strong theoretical basis that gives us the confidence that the approach is sound even if our implementation might be not yet that brilliant.
- The CIFAR10(10 classes)/CIFAR100(20 classes)/SVHN(11 classes) data sets are not space specific but are widely used synthetic class-sets and thus used to evaluate this concept.

Problem Statement

Q: What is needed so we can justify extrapolation from observed situations to yet unobserved situations ?

Thats in essence the ANN safety challenge - in control theory we can do that as inputs are (in general) a continuous range and outputs (ideally) related by a quantified function. Thus correctness of the control algorithm can be (with known constraints) guaranteed due to the underlying causal model — not so in current ANNs.

Constraint: this all only makes sense at the system level, treating elements individually (HW/SW,data/models) without having clear constraint sets makes little sense

Concept Space

- A highly anthropomorph approach
- Modeling the decision process along social choice and decision theory
- Using AI/ML as close to “ideal” human actors (due to the limits of social choice theory)
- Capitalizing on complexity to resolve or at least uncover uncertainty
- Striving for a quantitative results - but no quite there yet.
- Key assumption: testing in highly complex black-box systems is not suitable to achieve adequate assurance in open environments - never.

Hypothesis: Every sufficiently large set of complex black-box elements can be forced to behave statistically predictable - essentially just a crude variant of the law of large numbers.

Problem Space

- Relying on results of AI/ML for critical decision making in autonomous systems
- Assurance by testing only essentially — inherently insufficient
- In open-environments testing with a limited data set (size, features, variability and class-discriminating vs class-defining) is insufficient to give guarantees or even allow justified extrapolation.
- Single model issues:
  - Over-fitting
  - Inability to “know that they don’t know”
  - Data/scenario induced bias
  - Training/tuning induced bias
  - Model size -> resource demands and temporal impact
  - Limited/unknown random fault robustness of single models
  - A plethora of security issues (e.g. adversarial attacks)
- Impossibility-theorems of rule-based systems that are unable to express ”I don’t know”

Current “solution” space

- More data, more training, more validation, larger models...
- xAI
  - Inherent limits notably regarding (anthropomorph) interpretability
  - Labeling issue: Human perception works differently than machine perception
  - Unclear quantification
- single model issues
  - Best-practice (what ever that is exactly in an early phase of an evolving domain)
  - Not much theory behind it
  - Literally hundreds of parameters that have no defined valid range
- Certified processing elements ?
  - FIT values GPUs with GB or RAM ? not very realistic
  - Traditional gate-level analysis/FTA seems prohibitive anyway.
  - Again: testing of highly complex GPUs not very convincing and hardware is still in flux

Example: Classifiers - supervised learning

- Intent: detect class X in input image with a tolerable false-positive rate
- Training:
  - we train classifiers with many labeled images for a set of classes
  - The model learns to assign "probabilities" for a presented input which is the "probabilities" of matching these classes
  - We typically then select the result with the highest "probability" and assign the class label.
- Verification:
  - Testing against new images from the classes for which the model was trained (ID data).

"All models are wrong but some are useful" [George Box]

ANN decision uncertainty

- From a single instance we can not infer any uncertainty — but:
  - Without quantifiable uncertainty we can **not** build safe systems !
- Solution Spaces:
  1. Gain access to generalizable ML-elements distributions
  2. Force ML-elements into a robust system-level distribution

We believe that only the second strategy currently has any real chance of success.

Introducing Herds of Dumb Models

- Marquis De Condorcet (1785)
  - A large enough group of individuals each with $p_{correct} > .5$ will outperform any expert with $p_{correct} < 1$ - with other words all experts.
  - Hoeffding, Miller, Boland relaxed the strict $p_{correct} > .5$ later
- Unbiased models: generate random models - stacks of syntactically valid layers randomly selected, parameterized by random hyper-parameters within (empirically) sound boundaries.
- Train these models of a (randomly) selected subset of the training data replicated into sets and (randomly) augmented to form diverse "views" of the inputs.
- Train and forget: train the models in one run and store it - no tuning no (hidden) information leakage — thats as unbiased as we can get
- Train and forget also resolves the issue of generally not having new validation data available at some point

Removing Bias

Biased models are a safety problem as it constitutes an undetected **systematic** fault and hence does **not** have a probability attached that "protects" us. Given the wrong scenario it will reliably fail dangerously.

- Bias sources: labeling, model-design, parameter-tuning, model selection...
- Removing bias:
  - decouple model generation from specific problems
  - avoid any feedback cycles that may incur leakage
  - minimize the human error impact in all steps — ideally eliminate them notably because we can't quantify them (we probably can not even assign qualitative rankings)

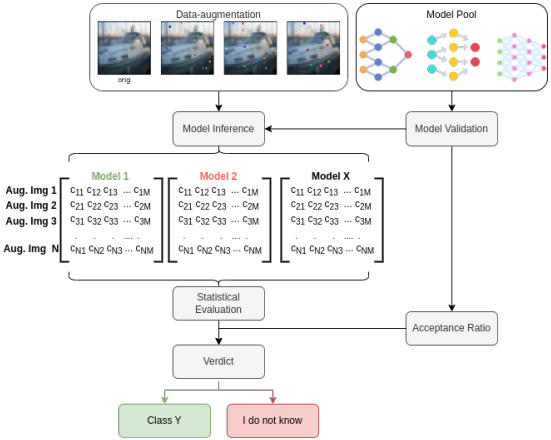
Statistics has a lot of methods for (ideally) removing bias problems — we will (mis)use some of them for ML.

A Bias-Free Model Hypothesis

- Hypothesis: **On average** ANNs trained on labeled inputs are better than coin-flips.
- Corollary: The single model is not trustworthy and no "individual" can reliably indicate its lack of "knowledge"
  1. Randomize the model architecture
  2. Randomize hyper-parameters
  3. Randomize the training sets
  4. Train against sets of images not individual images
  5. Randomize the inference machinery

If we need to avoid bias - randomization is a statisticians tool of choice The question left is how to get such a mess to do anything productive and notably deterministically ?

HDM (Herd of Dumb Models) - robust statistical approach - big-picture



- Concept: Build on social choice theory (Condorcets Jury Theorem) and statistical post processing
- Unbiased models: by randomizing layers, hyper-parameters, training data and model selection from the model pool during inference.

Preliminary Results

Early results indicate that the method works with un-tuned model sets as expected

Results ID: CIFAR-10 with 300 images sampled from test data set

Herds	1	2	3	4	5	6	7	8	Consensus
Total probs	300	300	300	300	300	300	300	300	300
Classified (TP)	175	177	176	173	178	172	173	180	134
Misclassified (FP)	2	2	3	3	8	2	3	1	0
I don't know	123	121	121	124	114	126	124	119	74

Results OOD: CIFAR-100 with 300 sampled images

Herds	1	2	3	4	5	6	7	8	Consensus
Total probs	300	300	300	300	300	300	300	300	300
Classified (TP)	0	0	0	0	0	0	0	0	0
Misclassified (FP)	44	45	47	42	45	40	38	42	15
I don't know	256	455	254	258	255	260	262	258	200

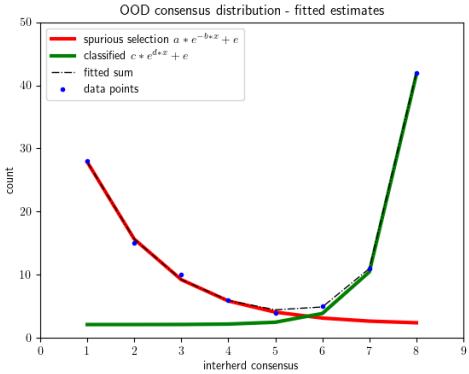
While the current proof-of-concept is not very efficient it is effective in the sense that if it claims the input is a truck then it is a truck (note that classes between CIFAR10 and CIFAR100 contain collisions e.g. trucks and vehicles2)

Inference with HDM

- Select a (random) set of N models from the pool
- Replicate the input R times and apply augmentation
- Submit set set of R replicas to all N models and collect the prediction vectors
- Statistically post process the stacked prediction vectors (prediction matrix) to assess the consistency of the herd.
- Do this with M herds and verify that they agree or declare “We don’t know”
- Quantify the failure rate by looking at the development over herds

Quantifying failure rates

- Employ multiple herds rather than one big herd
- Look at the consensus across herd-counts
- Model those as exponential sum of
  1. the decreasing spurious consensus
  2. the increasing systematic consensus



Current results

TODO:

ID inputs				FAR-OOD inputs			
- v9.3_m23_r40_b4_v24_2oo3.log0				- f9.3_m23_r40_b4_v24_2oo3.log0			
Total Inferences	:	1380000		Total Inferences	:	1380000	
Total Models tried	:	34500		Total Models tried	:	34500	
Total probs	:	500	100.0%	Total probs	:	500	100.0%
Classified	(TP):	360	72%	Classified	(TP):	0	0%
Misclassified	(FP):	0	0%	Misclassified	(FP):	9	1.8%
I don’t know	(TN):	14	2.8%	I don’t know	(TN):	178	35.6%
Unassured	(FN):	126	25.2%	Unassured	(FN):	313	62.6%

This is to be replaced by the final data once it is available

Limitations of the current results

- Filter optimization - we know that linear regressors are not optimal (but they are much easier to understand/debug at this point)
- Remaining heuristics need to be resolved
- Quantifying unbiasnes ? e.g. distribution over misclassified label assignments class association (ID and ID-OOD)
- Establishing robust criteria
- Unclear if the over-fitting impact is actually eliminated or not (just not enough data yet).
- Current memory usage is probably prohibitive for many space systems (on the other hand we might not need very robust storage for HDM)

Safety challenges

- Humans classify using causal-models and hence can infer data that is not present (and regularly do so) this leads to inconsistency of training data.
- Human perception differs significantly from data-driven machine perception — hence human labeled images are not automatically consistent for machines. A ML level "cleaning" is needed, I think, to have credible consistency of training data.
- Process issues: rigorous processes only make sense if a reasonable metrification allows effective PCDA cycles to be applied. At the current state of ML pipe-line handling this is not yet the case.
- Many of the standards on ML are in an early stage and only high-level conceptual parts available in a consolidated form. Hence we currently consider IEC 61508 Ed2 still as the best basis for system safety and notably guidance on processes.
- ML related hardware will never achieve credible FIT values and or bottom-up assurance. We think that the only path forward is to accept unassured hardware and to some extent software and derive the robustness/assurance at the system level.

Safety concept

- HDM is massive parallel small models with extensive randomization
- HDM uses majority voting and hence the likelihood of a false-positive
- Given the massive parallel nature - run it on unassured non-RAD-hard hardware and drop all those results that don't survive
- technically its a NooM(UooV) system that can be scaled to high robustness at the price of elevated false-negatives.
- Temporal aspects: the overall computation time, if fully parallel, may actually be lower; single scatter/gather + small independent models.
- Error detection time should be constant as we expect error detection to trigger at almost every inference at least on some replicas/models

This is though **conditioned** on the low-level mechanisms (e.g. dot()) being sufficiently robust.

Conclusions

- HDM is a, we believe, novel approach that allows to force ML into a statistically well-behaved realm
- This transition comes at the cost of inference on replicas as well as using many small (200k-2M trainable parameters) models
- Statistical robustness is what ultimately allows us to derive claims on OOD inputs as well as trustworthy failure rates
- The massive trivial parallel execution allows for robustness of the system even under assumption of high SEU rates.
- Statistical post-processing also allows to discover going out-of-scope and hence transit to a safe state
- Finally the approach allows to capitalize on complexity and help resolve the challenge of certifying complex hardware/software and ML-models.

Conclusions

- HDM is a, we believe, novel approach that allows to force ML into a statistically well-behaved realm
- This transition comes at the cost of inference on replicas as well as using many small (200k-2M trainable parameters) models
- Statistical robustness is what ultimately allows us to derive claims on OOD inputs as well as trustworthy failure rates
- The massive trivial parallel execution allows for robustness of the system even under assumption of high SEU rates.
- Statistical post-processing also allows to discover going out-of-scope and hence transit to a safe state
- Finally the approach allows to capitalize on complexity and help resolve the challenge of certifying complex hardware/software and ML-models.

The real conclusion: never trust the single expert....

Questions

Nicholas Mc Guire, <safety@osadl.org>
Imanol Allende, <imanol.allende@codethink.co.uk>
Carles Hernández: <carherlu@upv.es>