Automated Machine Learning

Cristobal Donoso Oliva

MIT-Chile Workshop - January 2023

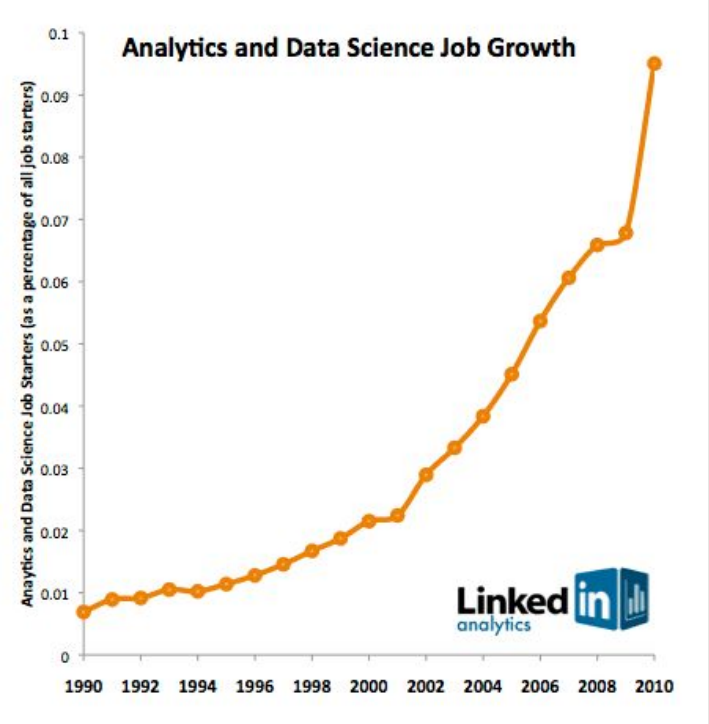
Motivation

Motivation

- Exploit data to create/improve products and services

Motivation

- Exploit data to create/improve products and services
- Business entities are hiring more and more data scientist and machine learning engineers



Motivation

- Exploit data to create/improve products and services
- Business entities are hiring more and more data scientist and machine learning engineers
- As stakeholders attempt to maximize the use of their data, automated machine learning became more relevant in Machine Learning research

Automated Machine Learning

Automated Machine Learning

A paradigm for **automating the application of machine learning** to real-world problems*

*we usually call it: **end-to-end process**

Karmaker, S. K., et.al, (2021). Automl to date and beyond: Challenges and opportunities. *ACM Computing Surveys (CSUR)*, *54*(8), 1-36.

Automated Machine Learning

A paradigm for **automating the application of machine learning** to real-world problems*

*we usually call it: **end-to-end process**

Karmaker, S. K., et.al, (2021). Automl to date and beyond: Challenges and opportunities. *ACM Computing Surveys (CSUR)*, *54*(8), 1-36.

End-to-end Machine Learning process

End-to-end ML process

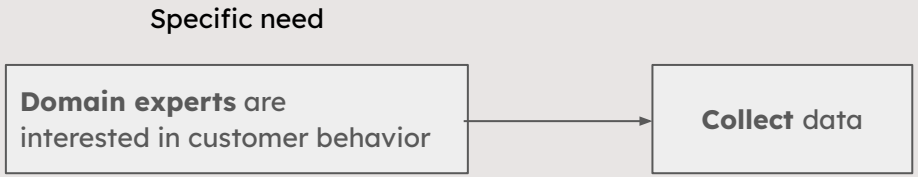
Specific need

End-to-end ML process

Specific need

Domain experts are
interested in customer behavior

End-to-end ML process



End-to-end ML process

Specific need

Domain experts are interested in customer behavior → **Collect** data → Get **insights from data**

End-to-end ML process

Specific need

Domain experts are interested in customer behavior

Collect **A LOT** of data

Get **insights from data**

End-to-end ML process

Specific need

Domain experts are interested in customer behavior

Collect **A LOT** of data

Get **insights from models**

Domain Expert vs Data Scientist

Domain Expert: A person who is fluent in the domain where ML is being applied but has minimal knowledge of how ML itself works

Data Scientist: A person who knows how ML works but has minimal knowledge of the domain where it is being applied.

End-to-end ML process

Specific need

Domain experts are interested in customer behavior

Collect **A LOT** of data

Get **insights from models**

End-to-end ML process

Specific need

Domain experts are interested in customer behavior → Collect **A LOT** of data → **Coordination** between Domain experts and data scientists

End-to-end ML process

Specific need

Domain experts are interested in customer behavior

Collect **A LOT** of data

Coordination between Domain experts and data scientists

Data scientist **understand** the context of the **problem**

End-to-end ML process

Specific need

Domain experts are interested in customer behavior

Collect **A LOT** of data

Coordination between Domain experts and data scientists

back-and-forth

Data scientist **understand** the context of the **problem**

Data scientist **translate the problem** to a **prediction task**

End-to-end ML process

Specific need

Domain experts are interested in customer behavior

Collect **A LOT** of data

Coordination between Domain experts and data scientists

back-and-forth

Data scientist **understand** the context of the **problem**

Data scientist **translate the problem** to a **prediction task**

Data scientist **draw valuable insights** from **data** and **models**

End-to-end ML process

Specific need

Domain experts are interested in customer behavior

Collect **A LOT** of data

Coordination between Domain experts and data scientists

back-and-forth

Data scientist **understand** the context of the **problem**

Data scientist **translate the problem** to a **prediction task**

Data scientist **draw valuable insights** from **data** and **models**

Domain experts and business leaders takes this conclusions to improve the product

AutoML

The term AutoML is commonly understood to denote a system that can perform the common data science tasks in an end-to-end process with minimal human intervention

End-to-end ML process

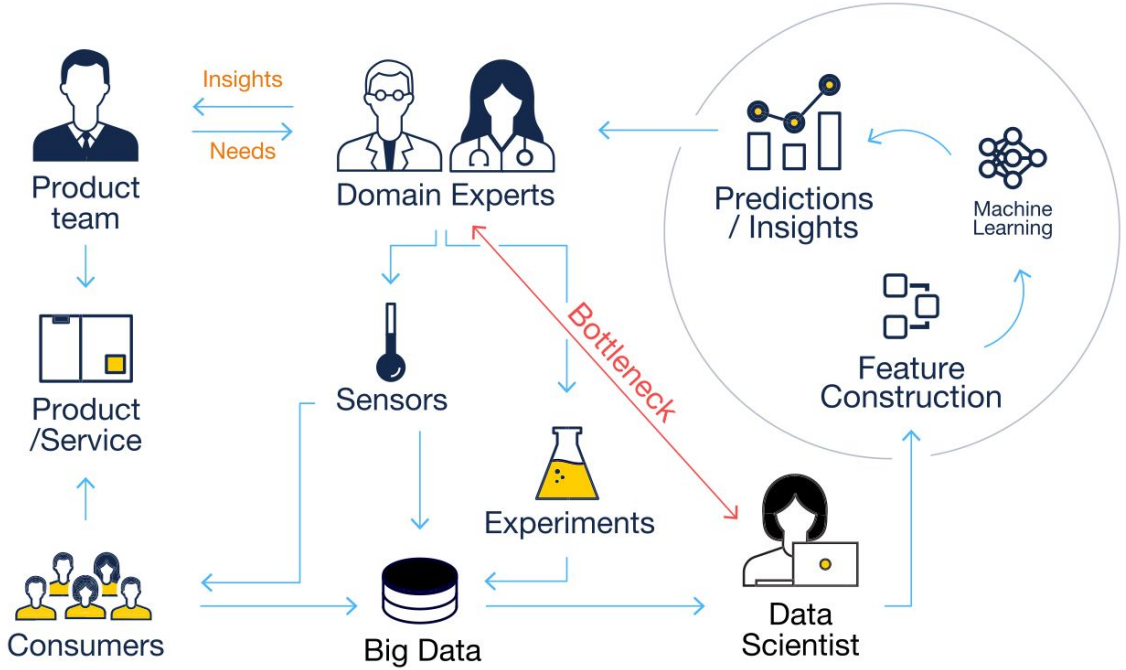


Fig. 1. A typical end-to-end machine learning process, with multiple stakeholders bringing their individual expertise, tools, methods, datasets, and goals.

Karmaker, S. K., et.al, (2021). Automl to date and beyond: Challenges and opportunities. *ACM Computing Surveys (CSUR)*, *54*(8), 1-36.

Levels of AutoML Systems

Level 0: Hand-coded implementation (ML researcher)

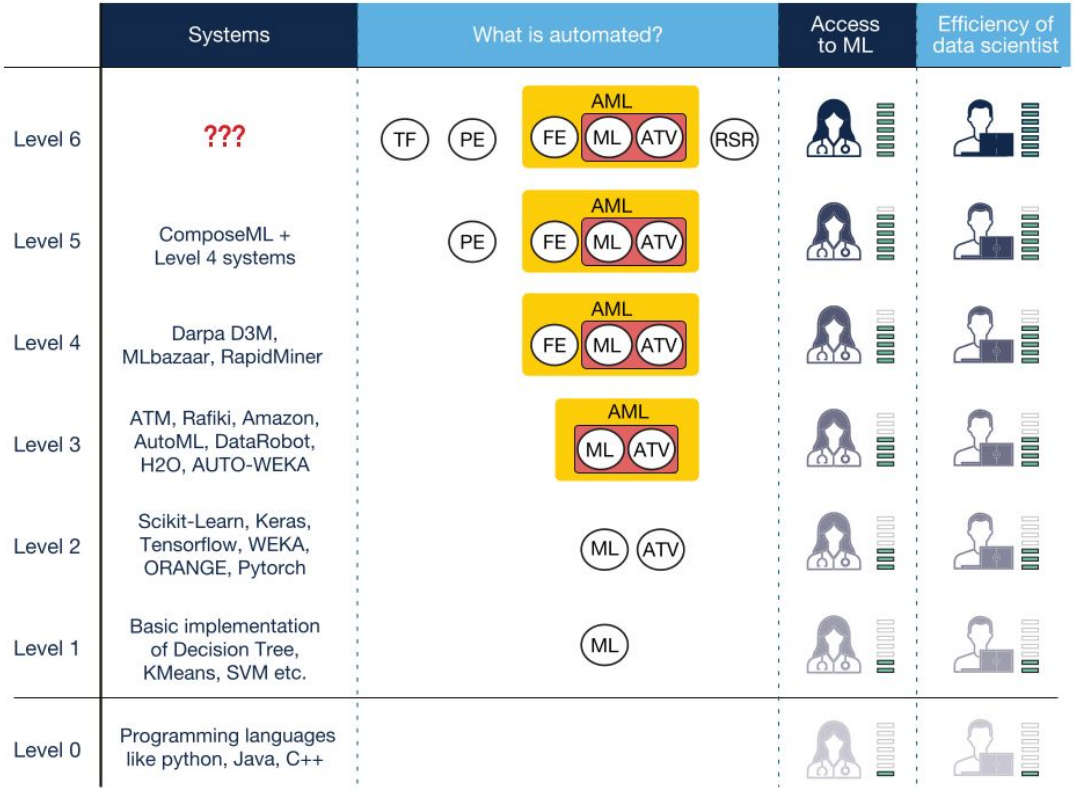
Level 1: C++ implementation of SVM classifier. Used as a library by data scientist.

Level 2: When training set is well-defined, data scientist can use software to perform the actual learning and parameter tuning tasks

Level 3: Like level 2 but ML and ATV together

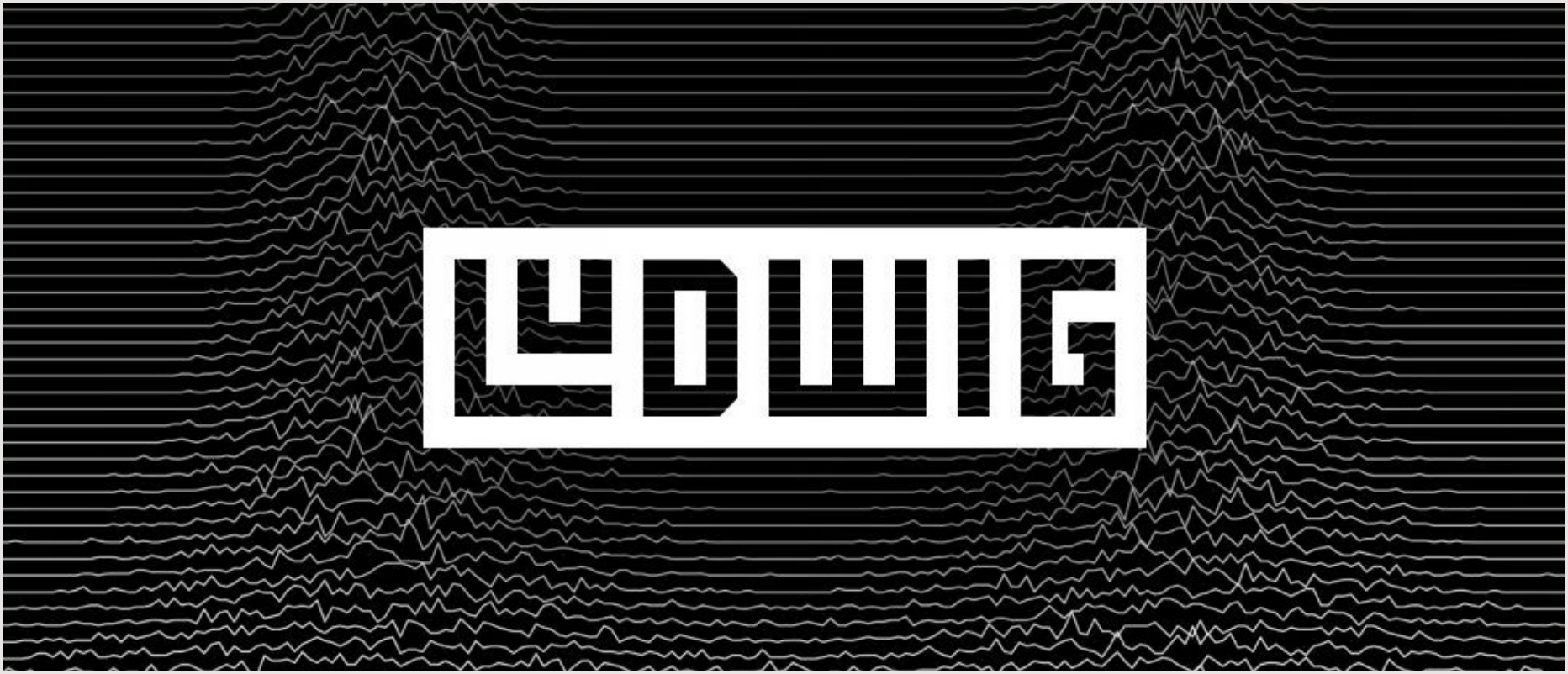
Level 4: Minimal interaction by domain experts

Level 5: Domain expert can comfortably interact with the system



Karmaker, S. K., et.al, (2021). Automl to date and beyond: Challenges and opportunities. *ACM Computing Surveys (CSUR)*, *54*(8), 1-36.

Hands-on!


LDWIG

Ludwig

Ludwig, an open source, deep learning toolbox built on top of TensorFlow that allows users to train and test deep learning models without writing code.

Ludwig

Ludwig, an open source, deep learning toolbox built on top of TensorFlow that allows users to train and test deep learning models without writing code.

Ludwig is unique in its ability to help make deep learning easier to understand for non-experts and enable faster model improvement iteration cycles for experienced machine learning developers and researchers alike.

Ludwig

Ludwig, an open source, deep learning toolbox built on top of TensorFlow that allows users to train and test deep learning models without writing code.

Ludwig [illegible] ster model i [illegible]

[illegible] We have witnessed its value to several of Uber's own projects, including our Customer Obsession Ticket Assistant (COTA), information extraction from driver licenses, identification of points of interest during conversations between driver-partners and riders, food delivery time prediction, and much more. [illegible]

Ludwig

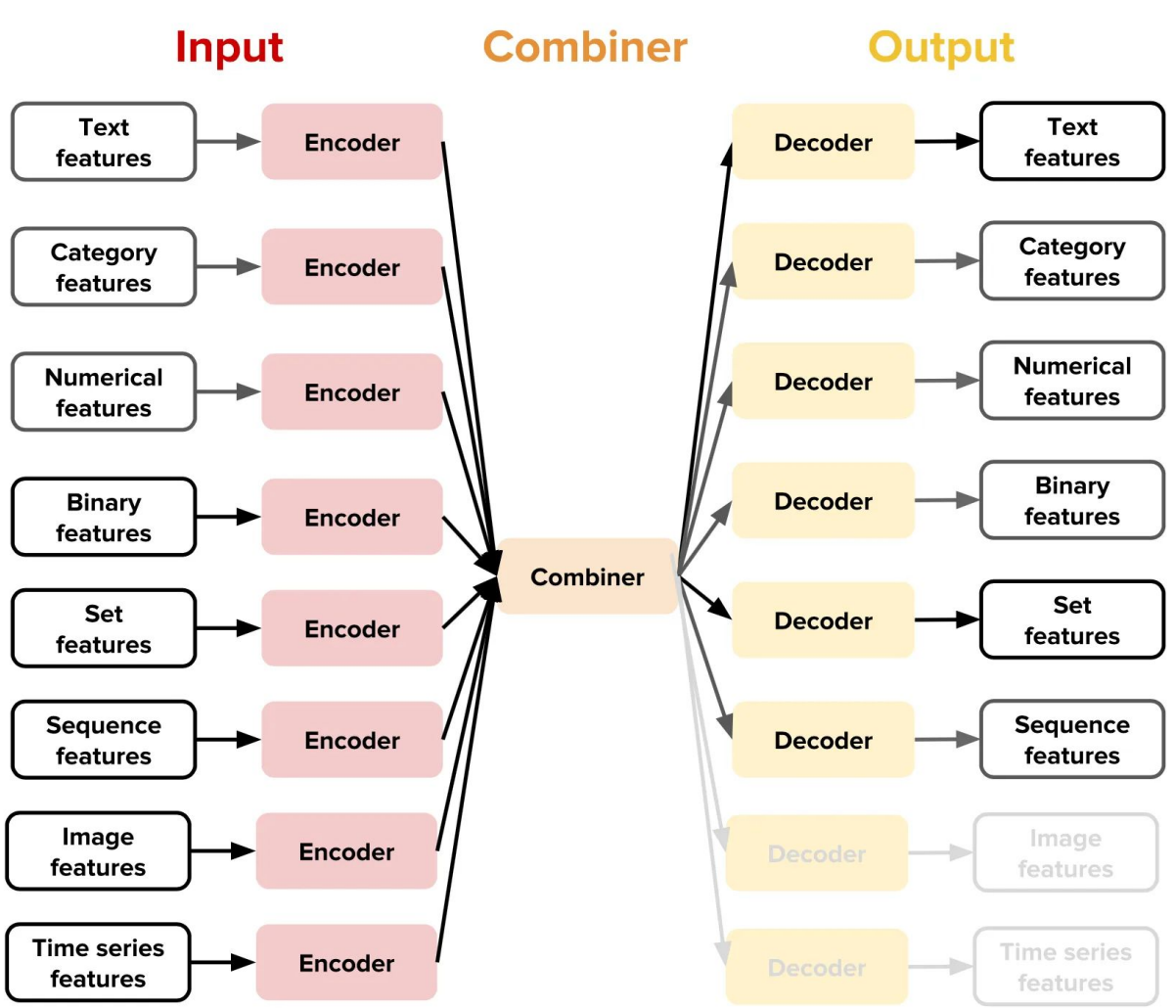
Ludwig, an open source, deep learning toolbox built on top of TensorFlow that allows users to train and test deep learning models without writing code.

Ludwig is unique in its ability to help make deep learning easier to understand for non-experts and enable faster model improvement iteration cycles for experienced machine learning developers and researchers alike.

If deep learning libraries provide the building blocks to make your building, Ludwig provides the buildings to make your city, and you can chose among the available buildings or add your own building to the set of available ones.

Ludwig

Data type-specific **encoders** and **decoders**

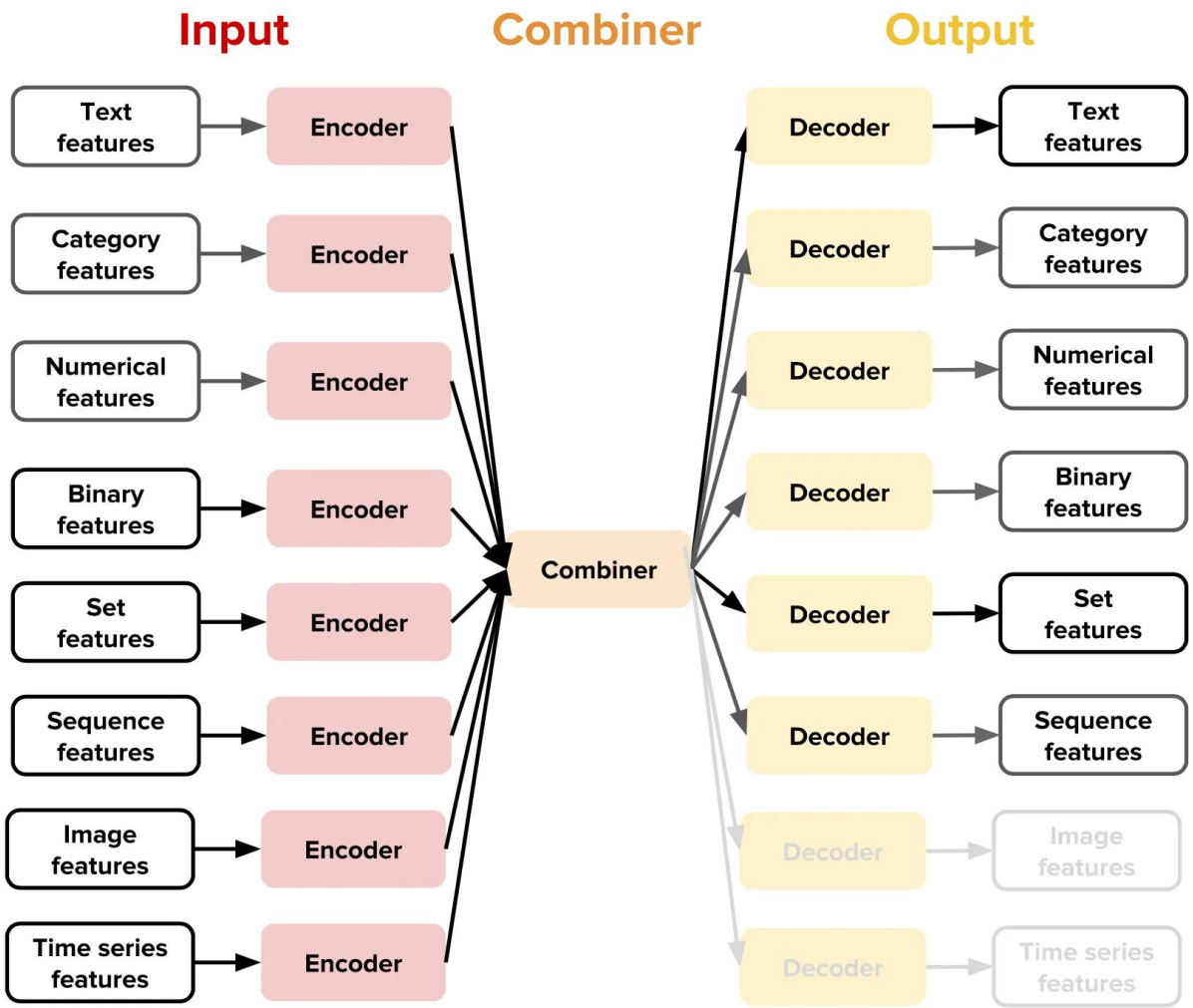


Piero Molino et.al., (2019) Introducing Ludwig, a Code-Free Deep Learning Toolbox

Ludwig

Data type-specific **encoders** and **decoders**

Each data type has a specific processing function



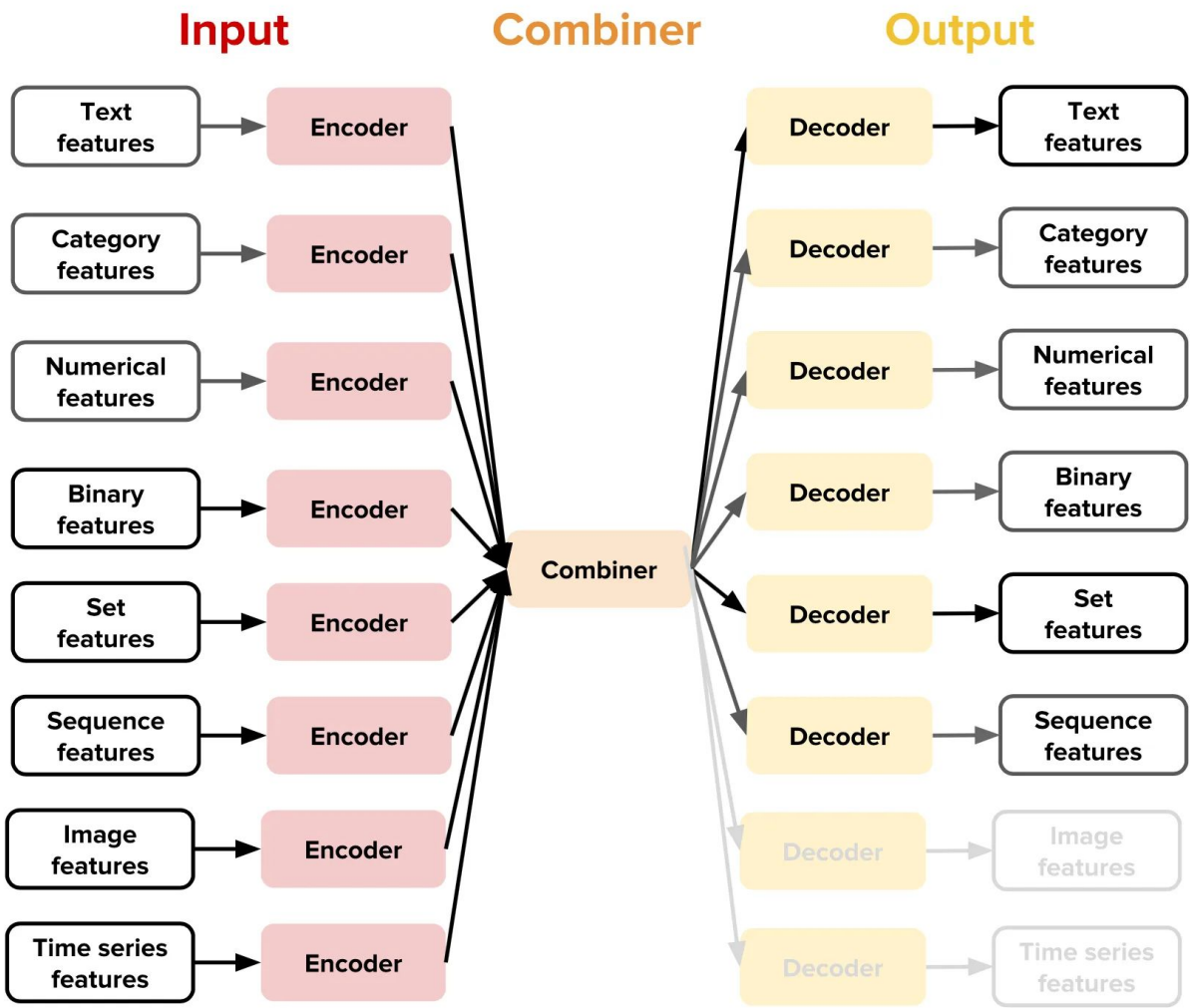
Piero Molino et.al., (2019) Introducing Ludwig, a Code-Free Deep Learning Toolbox

Ludwig

Data type-specific **encoders** and **decoders**

Each data type has a specific processing function

Ludwig incorporates a set of command line utilities for training, testing models, and obtaining predictions



Piero Molino et.al., (2019) Introducing Ludwig, a Code-Free Deep Learning Toolbox

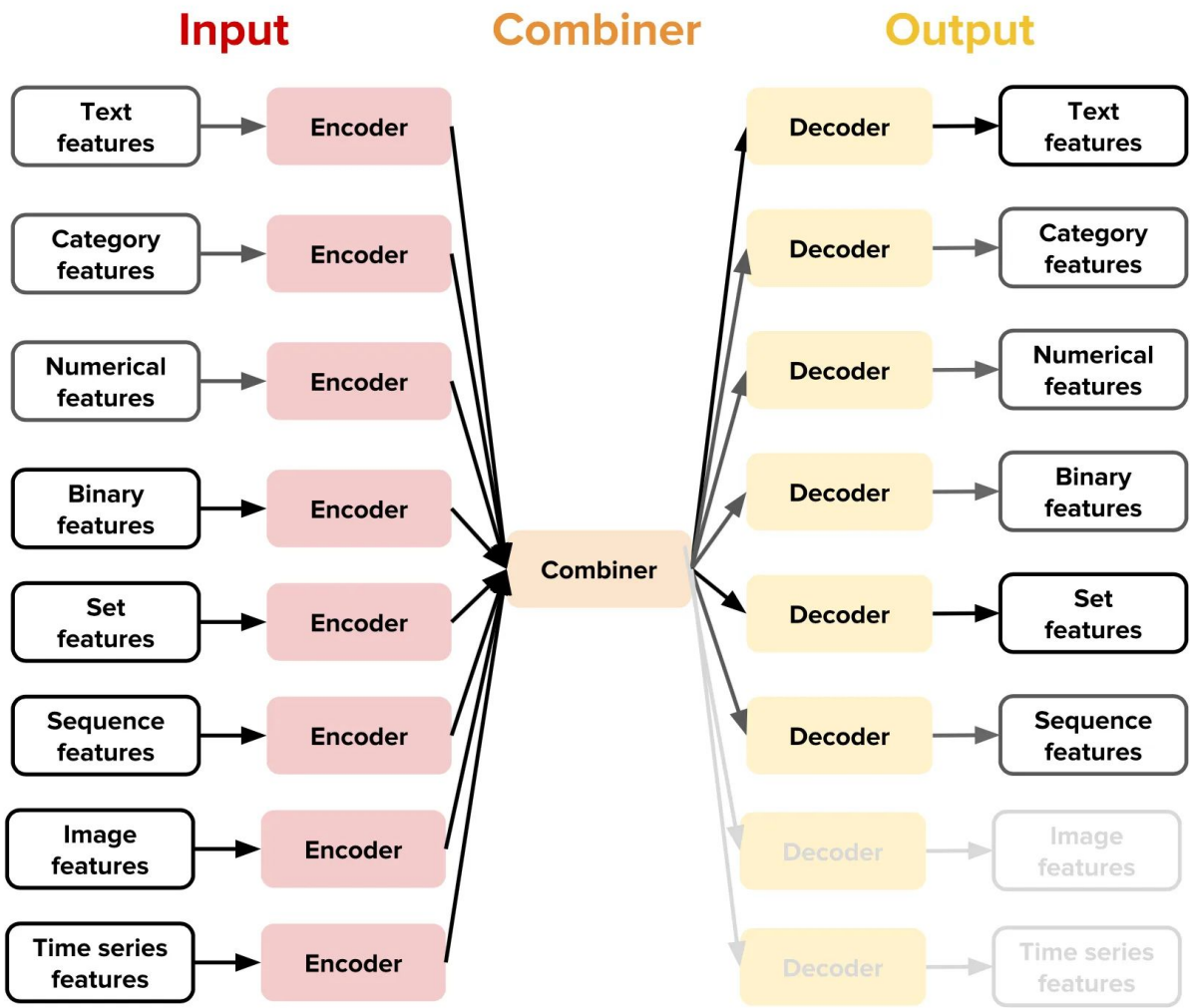
Ludwig

Data type-specific **encoders** and **decoders**

Each data type has a specific processing function

Ludwig incorporates a set of command line utilities for training, testing models, and obtaining predictions

Train models on multiple GPUs locally and in a distributed fashion through the use of Horovod



Piero Molino et.al., (2019) Introducing Ludwig, a Code-Free Deep Learning Toolbox

Installation

Python (with Pip) recommended

```
pip install ludwig
```

This will install Ludwig's basic requirements for modeling with binary, category, number, text, image, and audio features. The requirements for additional functionality are separated out so that users are able to install only the ones they actually need:

- `ludwig[serve]` for serving dependencies.
- `ludwig[viz]` for visualization dependencies.
- `ludwig[hyperopt]` for hyperparameter optimization dependencies.
- `ludwig[distributed]` for distributed training on Ray using Dask and Horovod.

The full set of dependencies can be installed with:

```
pip install 'ludwig[full]'
```

GPU support

If your machine has a GPU to accelerate the training process, make sure you install a GPU-enabled version of PyTorch before installing Ludwig:

```
pip install torch -f https://download.pytorch.org/whl/cu113/torch_stable.html
```

The example above will install the latest version of PyTorch with CUDA 11.3. See the official PyTorch docs for more details on installing the right version of PyTorch for your environment.

https://ludwig.ai/latest/getting_started/installation/

Installation

Docker

The Ludwig team publishes official Docker images that come with the full set of dependencies pre-installed. You can pull the `latest` images (for the most recent official Ludwig release) by running:

```
docker pull ludwigai/ludwig:latest
```

The `ludwig` command line tool is provided as the entrypoint for all commands.

GPU support

If your machine has a GPU to accelerate the training process, pull the official Ludwig image with GPU support:

```
docker pull ludwigai/ludwig-gpu:latest
```

Python (with Pip) recommended

```
pip install ludwig
```

This will install Ludwig's basic requirements for modeling with binary, category, number, text, image, and audio features. The requirements for additional functionality are separated out so that users are able to install only the ones they actually need:

- `ludwig[serve]` for serving dependencies.
- `ludwig[viz]` for visualization dependencies.
- `ludwig[hyperopt]` for hyperparameter optimization dependencies.
- `ludwig[distributed]` for distributed training on Ray using Dask and Horovod.

The full set of dependencies can be installed with:

```
pip install 'ludwig[full]'
```

GPU support

If your machine has a GPU to accelerate the training process, make sure you install a GPU-enabled version of PyTorch before installing Ludwig:

```
pip install torch -f https://download.pytorch.org/whl/cu113/torch_stable.html
```

The example above will install the latest version of PyTorch with CUDA 11.3. See the official PyTorch docs for more details on installing the right version of PyTorch for your environment.

Example

title	author	description	cover	genre	price
Do Androids Dream of Electric Sheep?	Philip K. Dick	By 2021, the World War has killed millions, driving entire species into extinction and sending mankind off-planet. ...	path-to-image/do-android-cover.jpg	sci-fi	9.32
War and Peace	Leo Tolstoy	War and Peace broadly focuses on Napoleon's invasion of Russia in 1812 and follows three of the most well-known characters in literature...	path-to-image/war-and-peace-cover.jpg	historical	5.42
The Name of the Rose	Umberto Eco	In 1327, Brother William of Baskerville is sent to investigate a wealthy Italian abbey whose monks are suspected of heresy. ..	path-to-image/name-of-the-rose-cover.jpg	historical	16.99

csv tabular data

Piero Molino et.al., (2019) Introducing Ludwig, a Code-Free Deep Learning Toolbox

Example

title	author	description	cover	genre	price
Do Androids Dream of Electric Sheep?	Philip K. Dick	By 2021, the World War has killed millions, driving entire species into extinction and sending mankind off-planet. ...	path-to-image/do-android-cover.jpg	sci-fi	9.32
War and Peace	Leo Tolstoy	War and Peace broadly focuses on Napoleon's invasion of Russia in 1812 and follows three of the most well-known characters in literature...	path-to-image/war-and-peace-cover.jpg	historical	5.42
The Name of the Rose	Umberto Eco	In 1327, Brother William of Baskerville is sent to investigate a wealthy Italian abbey whose monks are suspected of heresy. ..	path-to-image/name-of-the-rose-cover.jpg	historical	16.99

csv tabular data

```yaml
input_features:
 -
   name: title
   type: text
 -
   name: author
   type: category
 -
   name: description
   type: text
 -
   name: cover
   type: image
output_features:
 -
   name: genre
   type: category
 -
   name: price
   type: numerical
training:
 epochs: 10
```

YAML metadata file

Piero Molino et.al., (2019) Introducing Ludwig, a Code-Free Deep Learning Toolbox

Example

Train

```
ludwig train –data_csv path/to/file.csv –model_definition_file
model_definition.yaml
```

Prediction

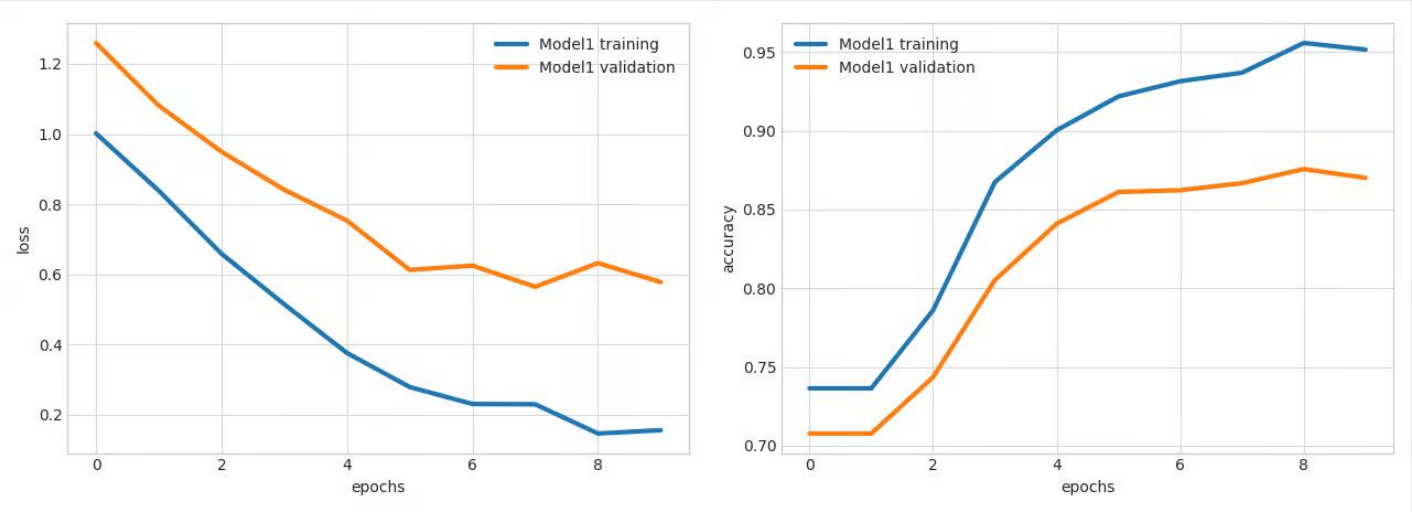
```
ludwig predict –data_csv path/to/data.csv –model_path /path/to/model
```


Piero Molino et.al., (2019) Introducing Ludwig, a Code-Free Deep Learning Toolbox

Example

Visualize

```
ludwig visualize –visualization learning_curves –training_stats
results/training_stats.json
```

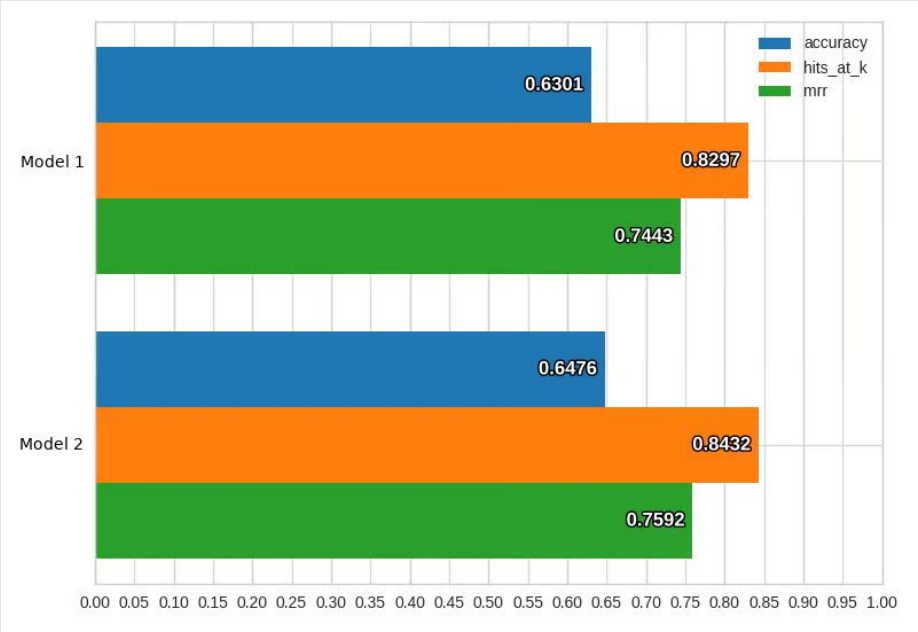


Piero Molino et.al., (2019) Introducing Ludwig, a Code-Free Deep Learning Toolbox

Example

Compare models

```
ludwig visualize –visualization compare_performance –test_stats
path/to/test_stats_model_1.json path/to/test_stats_model_2.json
```



Piero Molino et.al., (2019) Introducing Ludwig, a Code-Free Deep Learning Toolbox

Hands-on!

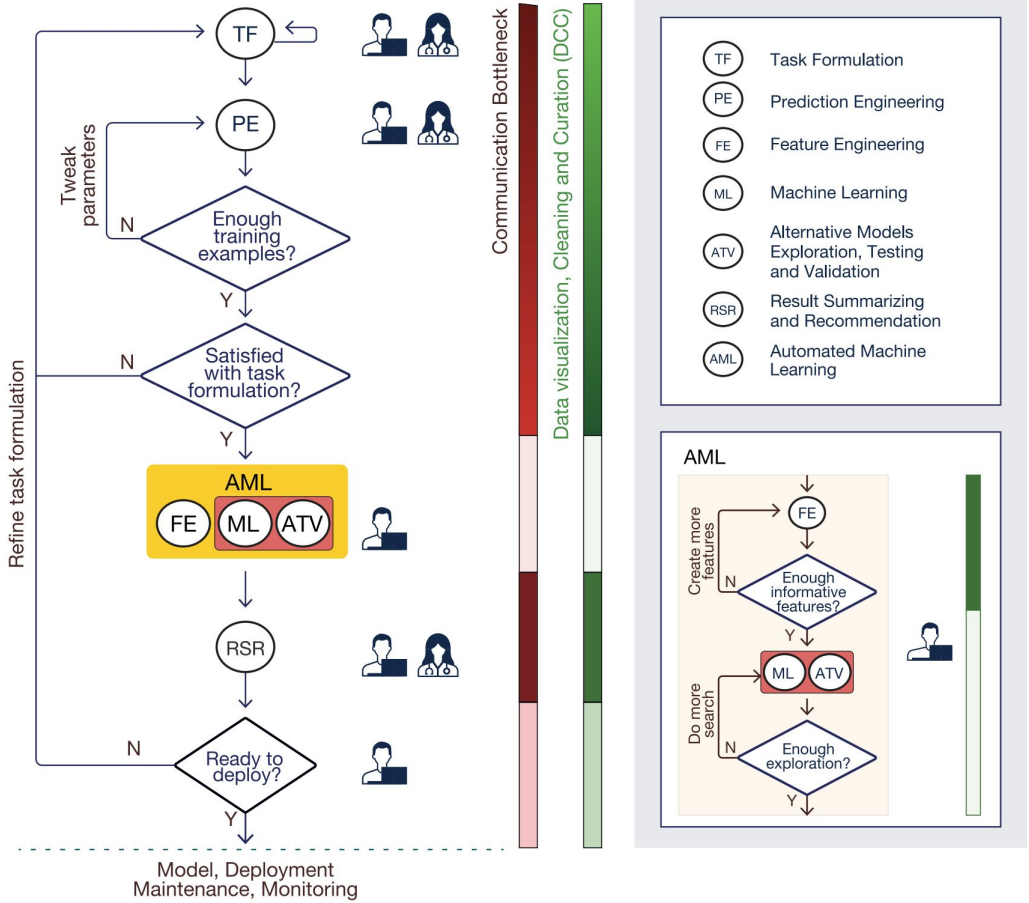


Fig. 2. A flowchart showing the machine learning process. This chart highlights points of interaction between domain experts and data scientists, along with bottlenecks.

Automated Machine Learning

Cristobal Donoso Oliva

MIT-Chile Workshop - January 2023

State of the Art: Model Generation

State of the Art: Model Generation

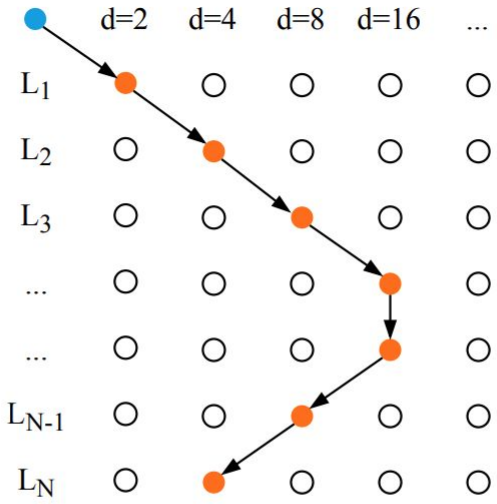


Figure 9: Network-level search space proposed by [129]. The blue point (top-left) indicates the fixed "stem" structure, the remaining gray and orange points are cell structure, as described above. The black arrows along the orange points indicate the final selected network-level structure. "d" and "L" indicate the down sampling rate and layer, respectively.

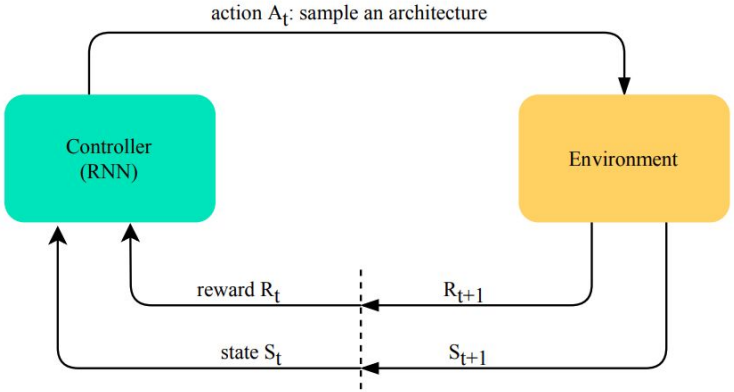


Figure 14: Overview of neural architecture search using reinforcement learning.

He, X., Zhao, K., & Chu, X. (2021). AutoML: A survey of the state-of-the-art. *Knowledge-Based Systems*, *212*, 106622.