The Business Executive's Guide to

SMART APPLICATIONS

GETTING STARTED WITH SMART APPLICATIONS & MACHINE LEARNING

UrulT

Machine learning, Artificial Intelligence; these are no longer simply "out there" concepts that provide the subject matter of Sci-Fi novels and comic books: they're quickly transforming the technology that we use on a day to day basis.

Every appearance of these new technologies, from the most obvious ones like the up-and-coming self-driving cars, to the more common forms such as streaming apps like Spotify and Netflix, is evidence that we have entered a new era.

In this ebook, you will find insights into what Smart Applications are, the benefits they offer, and information about Machine Learning's state-of-the-art technologies. You will find out how to combine classic approaches like Information Retrieval Systems (IRS) techniques with Recommender Systems (RS) and how to use good practices in the user interface design and user experience (UX) to transform traditional information systems into smart ones.

Any company that creates systems or applications will eventually have to adopt one "smart" technology or another, lest go the way of the floppy disk! So let's dive in!

> With machine learning a system could benefit from predictions and preferences to implement suggestions, maximize opportunities, minimize risks and take action.

SMART SYSTEMS - LET’S GET INTO SOME DEFINITIONS FIRST

PREDICTIONS

Some of the fundamental concepts when implementing intelligent systems are: **prediction, predictive models** and **predictive analytics.**

The above are related to the creation of math and statistical models which link observed values for a variable of interest (sometimes called the dependent or response variable) with contextual information, in other words, the data available (also called predictors or independent variables which explain what was observed).

EXPERT´S CORNER

Many complementary techniques like Data mining, dimentionality reduction, feature selection/extraction (among others) are used to find, clean and transform data to get the best predictors.

Dimensionality Reduction: the action of simplifying the complexity of the problem.

Some of the most common techniques are:

- Principal Components Analysis (PCA)
- Linear discriminant analysis (LDA)
- Decision Trees
- Random Forests
- Factor Analysis
- Partial Least Squares (PLS) and
- t-Distributed Stochastic Neighbor Embedding (t-SNE)
- Recommendation System techniques
  - Matrix Factorization
  - Single Value Decomposition (SVD)

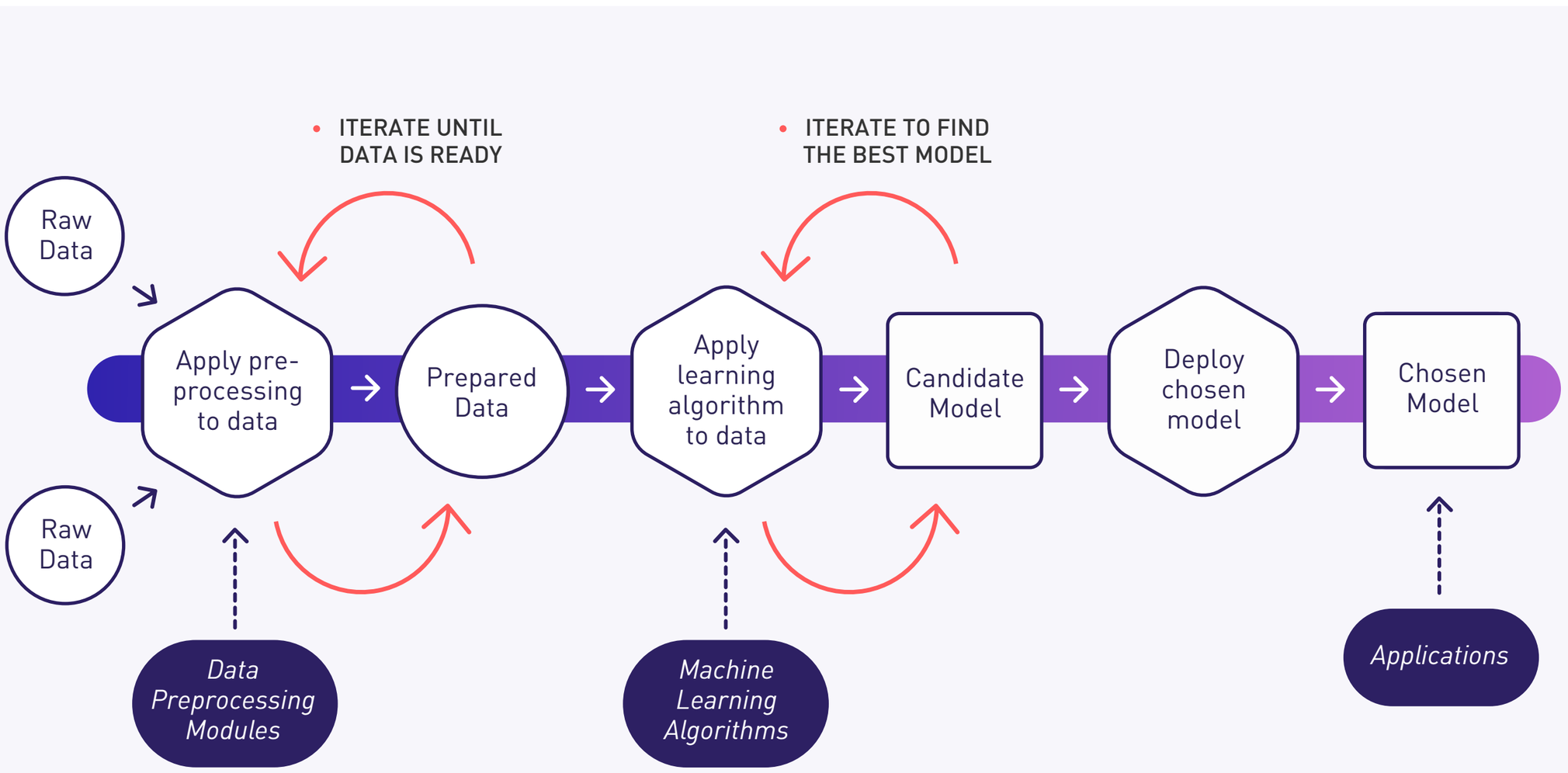
LEARN MORE: → **Beginners Guide To Learn Dimension Reduction Techniques** by S. Ray, K. Jain, & D. Gupta

MACHINE LEARNING

Smart Systems are strongly based on predictions, which are usually generated from:

- The recognition of patterns in data, which many times come from multiple heterogeneous sources and need preprocessing or transformation towards a common model.

- The selection and extraction of the most significant characteristics which allow reducing the complexity of the problem and relating it in some way to generate a predictive model. This will then be deployed and used later by an application or system in order to maximize opportunities or minimize risks (among other things).



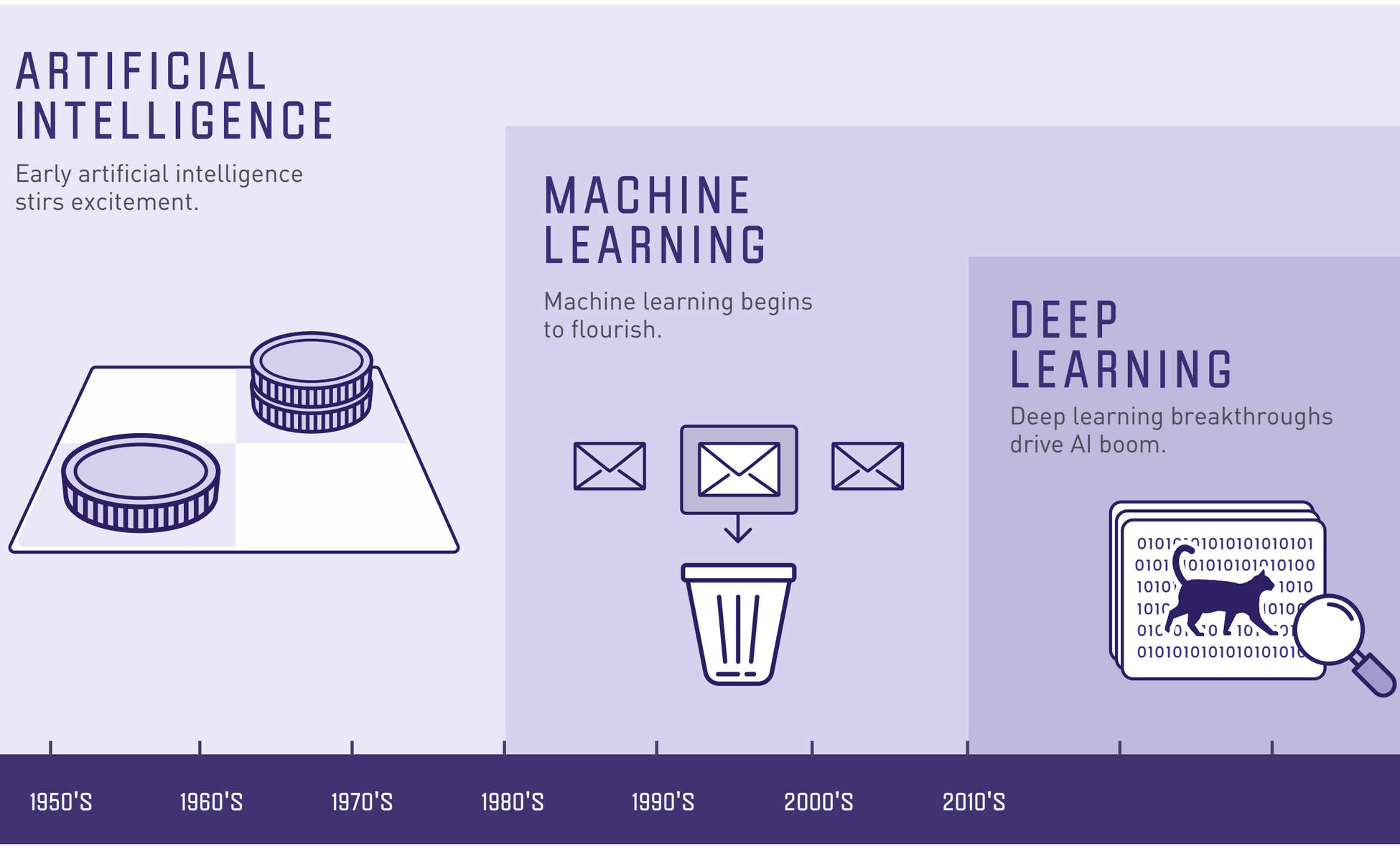
The Machine Learning Process - D. Chappell, "Introducing Azure Machine Learning - A guide for technical professionals," David Chappell & Associated - White Papers. Download

A BIT OF HISTORY

Machine Learning (ML) is a term that has become popular in recent years although it has existed for many decades, mainly due to the progress made in **Artificial Neural Networks (ANN)** and **Deep Learning (DL).** These are **specific techniques** from Machine Learning that have enabled us to move significantly forward in cognitive areas such as: **image processing, text analysis** or **natural language processing (NLP).**

In the information age, the **Internet of Things (IoT)** and **Big Data** have **empowered** the development of *Machine Learning* and smart applications, due to their great capacity to generate, process and store large amounts of data. It is **the fuel** that *Machine Learning* needs in order to work.

Moreover, *Machine Learning* is a field inside **Artificial Intelligence (AI).**



M. Copeland, "The Difference Between AI, Machine Learning, and Deep Learning? | NVIDIA Blog," The Official NVIDIA Blog, 29-Jul-2016. Download

Just like people learn and improve their actions through experience and information, programs based on *Machine Learning* improve their results accuracy through the constant use of the system, the data from the interaction of the users and the contextual information obtained from multiple external sources.

TRAINING THE MACHINE

The algorithms from Machine Learning can **be supervised** or **unsupervised**, depending on their training mode.

SUPERVISED

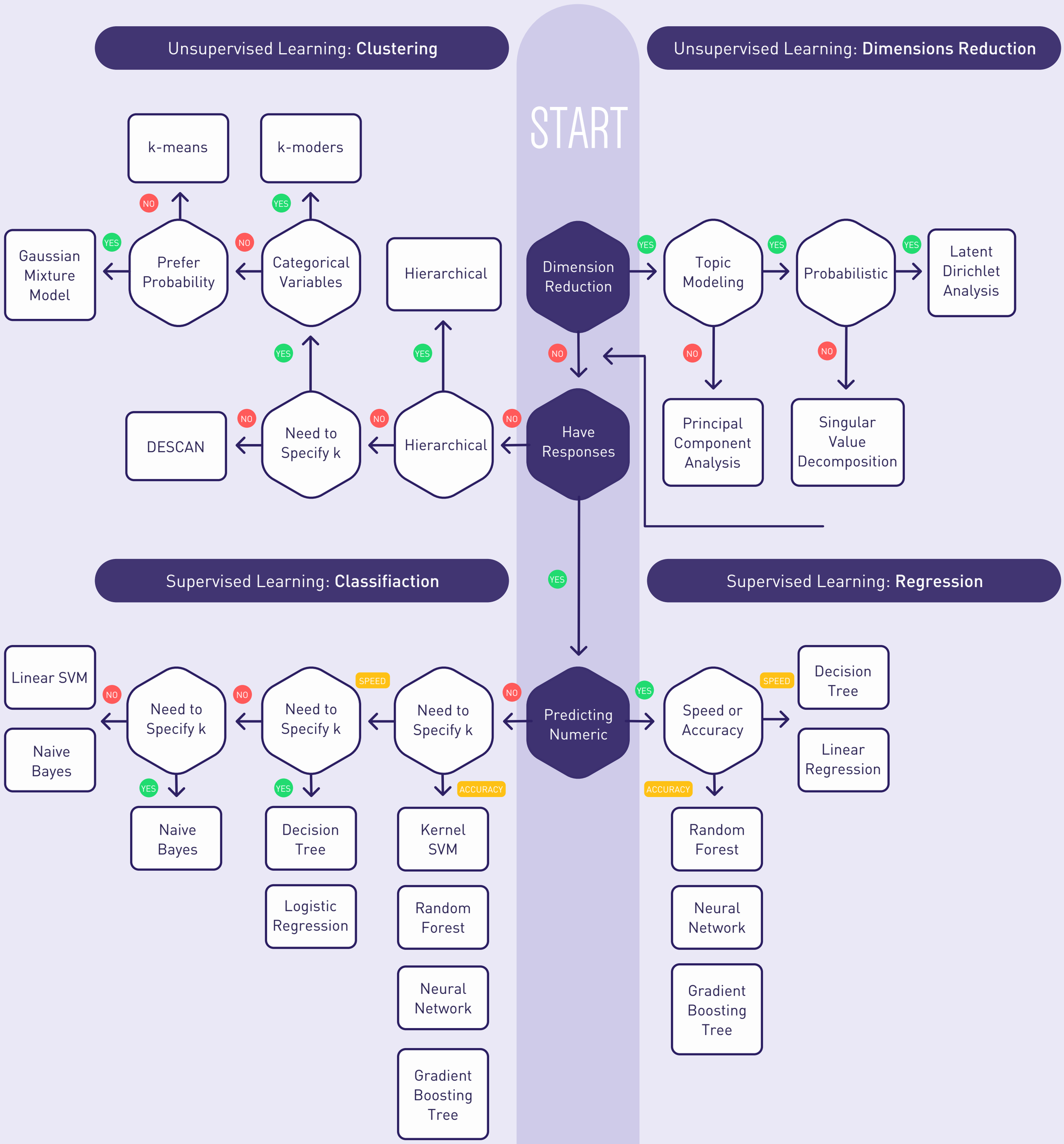
The algorithm is **supervised** if, when training our model, for each input we indicate the expected output – also called label, target or response. In this case, if the output has a numeric value among an infinite amount of possible values, we are talking about a **regression** problem. On the contrary, if the value is discrete or finite, each output is called "class" and the problem to solve is a **classification** problem. If there were only two outcomes (yes/no, true/false, etc.) the problem would be called **binary classification.**

UNSUPERVISED

In contrast, the algorithm is **unsupervised** if, when training our model, the expected outputs for the different inputs are not taught. In this case, the unsupervised algorithms are useful to understand the structure and the organization of the data. They are usually used to group the inputs into groups or clusters **(clustering)** or to apply **dimensionality reduction**.
Unsupervised algorithms are difficult to evaluate because there are no expected results with which to compare a prediction.

The diagram below shows some of the main algorithms of Machine Learning and in which category they could be employed. We can see that some may be used in more than one category.

·MACHINE LEARNING ALGORITHMS CHEAT SHEET·



Source: H. Li, "Which machine learning algorithm should I use? - Subconscious Musings," Subconscious Musings, 12-Apr-2017. Download

BONUS TRACK: Reinforcement Learning (RL) is another approach that differs from supervised or unsupervised learning and it has a special importance in the AI field. When using RL techniques, the learning phase is built by trial-and-error, rewards or punishments.

RECOMMENDER SYSTEMS

A Recommender System tries to predict users' system preferences to make suggestions. The personalization of the advertisement in a news website or the estimation of the kind of news to suggest to a specific user are typical problems that can be solved with recommender systems.

EXPERT´S CORNER

The most used techniques:

> collaborative filtering

> content-based filtering

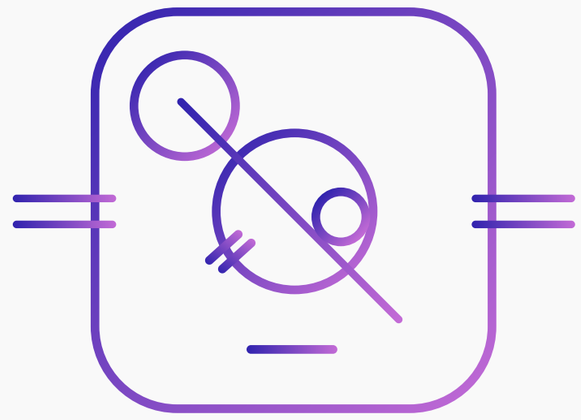
> Matrix Factorization

However, the *Netflix Price* competition has shown that a **Hybrid approach**, which combines different techniques, is the one that achieves better results

LEARN MORE:

→ **Recommender Systems Handbook** by F. Ricci, L. Rokach, and B. Shapira

→ **"Let´s go to the cinema! A movie recommender system for ephemeral groups of users,"** by Guillermo Fernández, Waldemar López, Bruno Rienzi, Pablo Rodríguez-Bocca, CLEI Electronic Journal. [Online]. Available: http://www.clei.org/cleiej/paper.php?id=338. [Accessed: 23-May-2017].



UNDERSTANDING THE POWER OF SMART APPS

SMART APPLICATIONS

CONTENT PERSONALIZATION

Let's imagine a news website that learns from our past interactions with the system: the **past** news we read, the **rating** we give (explicit feedback), the amount of time we dedicate to reading (implicit feedback), the semantics of what we write in the comments of the news (also **implicit feedback**), or how **similar users** act.

Google News or News360

SMART

1. **Picks** news to show in the main section.
2. **Adjusts and personalizes user experience** (layout, colors, font size, help, etc.) based on the user's profile: age, sex, nationality, educational level, etc., context: day and hour of the week or emotions that the user shared in his/her social network.
3. Personalizes the **advertising** shown to each user.
4. Analyzes marked groups/segments with similar interests to send the weekly press summary via email.
5. Based on information such as hour and day of the week, type and impact of news, predicts the number of visits during the next few hours. This will allow the site to be prepared for possible peak swipe hours in the system and to automatically launch more nodes from the web farm when needed.

Analyzing the profile of the users to show them specific content is a technique employed by **on-demand content streaming services and personalized ad or marketing campaigns**. Besides the content, after analyzing the user's profile, we can personalize other things, from the type of help shown, the type of font, colors, etc., up to the organization of the layout on the page.

RECOMENDATIONS

Nowadays, systems based on recommendations or suggestions can be seen anywhere: from streaming services such as Spotify or Netflix, e-commerce platforms such as eBay and Amazon, up to an endless number of applications which search for better **matches between users and items.**

You can find examples of matches in applications of the **Information Retrieval (IR)** area, which arrange the search results taking into consideration the items which will most likely be of greater preference to the user.

Another example is in **Human Resources** which aims to match the best candidates with a position or serve up the best suggestions for finding a job.

DOCUMENT/TEXT CLASSIFICATION

Classifying emails as spam or not, **recognizing** the language of a text, determining if a comment is positive or negative (**sentiment analysis**), and identifying if a text was written or not by a specific person are only some daily examples in which techniques of Machine Learning have been used to solve text classification problems. In the news website example, the sentiment analysis can be used to process our users' comments and obtain implicit feedback, which can be used as input for the engine's model that processes the recommendations.

OPERATIONAL IMPROVEMENTS IN THE BANKING SECTOR

Machine Learning techniques are also used in banking. In this area, these techniques deal with problems such as **fraud detection**, predicting **customer churn** and even the optimization of cash in ATMs.

Recently, we carried out a study for UruIT's customer relationship management business unit with an analysis of these and other opportunities to apply predictive analysis in the Banking area.

MARKET PRICE ESTIMATION

Estimating the real value of a product on the market is another interesting field of application for Machine Learning. The problem here is predicting the value of the product due to its characteristics and the present and future contexts.

A specific example is the assessment of immovable property due to location, years of construction, number of rooms, economic projections, etc. In its most simple way this could be set as a linear regression problem, but achieving highly efficient predictions could be more complex and by far more interesting to solve. So much so, that the popular platform Kaggle (recently included in Google Cloud) competes with "Zillow Prize: Zillow's Home Value Prediction (Zestimate)" sponsored by Zillow Inc. In this competition, the data science community competes for a one million dollar prize to find

the ideal algorithms to improve the precision of the estimator, Zestimate, and evolve its predictions about future sales prices in real estate. Would you try it?

FACE AND IMAGE RECOGNITION

Facial recognition or identification and image recognition is maybe one of the best known applications of deep learning. It's achieved by using a special type of artificial neural network (ANN), inspired by neuronal connections of the visual cortex known as the Convolutional Neural Network (CNN).

An example of the use of these systems can be seen in Uruguay, where the neuronal network is trained to recognize the face of violent individuals from a database and trigger an event to detect a potentially dangerous situation. After developing this technology, a security system was launched in 2017 to deny access to violent people in soccer stadiums through facial identification.

Other examples of these types of applications are the recognition of text in images with Optical Character Recognition (OCR) techniques, Facebook's tag suggestions, Google SafeSearch filter of pornographic images or the new feature of Similar Items to get information or buy products from Google Images search results.

SPEECH RECOGNITION

Some common tasks in which speech recognition is applied are dictating to a text processor, controlling computers or phones through voice commands, correcting pronunciation through language assistant, detecting if a voice belongs to a person or not, etc. Even though this is nothing new, huge progress was made in August 2017, when Microsoft Research managed to achieve a 5.1% error level, which is equal to the human error rate. This is the result of many years of effort from both the academic community and the industry. It represents a true milestone for Artificial Intelligence.

ANOMALY DETECTION

Machine Learning techniques for anomaly detection are used in several fields to detect “strange” patterns in data. Fraud detection, network intrusions detection, monitoring systems, amongst other things, are some of their most common uses.

These techniques can also be used during the pre-processing data stage before the execution of supervised learning to eliminate anomaly entries that could harm the model’s precision learned.

LEARN MORE: → The article in the **IBM blog** and the course video about *Machine Learning* **by Coursera** are a good starting point for the overfitting problem. This **paper** is also a good reference if you want more information about anomaly detection.

MACHINE LEARNING EVERYWERE!

We can find examples or opportunities to apply Machine Learning in almost any field that pops up in our minds. For example:

- In medicine, to detect cancer or make a diagnosis
- In education, to personalize learning content for students to predict grades
- In public security, to predict crimes
- In technology security to predict threats
- In Chatbots or customer support
- In the automotive industry, already in production in Self driving cars
- In entertainment, to create movie trailers

And many, many others! That's why **Machine Learning is what everybody talks about!**

FOR TECHIES:
STARTING TO IMPLEMENT
SMART SOLUTIONS

SMART APPLICATIONS

From a technology standpoint, there are two options: we can choose **cloud services** for machine learning or **specialized frameworks** which could be embedded directly into our applications.

CLOUD SOLUTIONS

Cloud solutions work quite well in general; we don't need to know a lot to start using them. Many times we have visual aids which help create our models, some can be used for free and only require paying once the usage increases.

Due to the cloud feature one of its main advantages is that it provides a highly scalable and safe infrastructure. Not only does it offer the possibility to create predictive models to solve common problems, but it also provides specific services to work in cognitive areas such as image processing or natural language. This simplifies the task of adding features such as face or object detection or voice recognition. In this way, we aim to maximize the user experience in our application.

FRAMEWORKS, LIBRARIES & LANGUAGES

If we need to manipulate our algorithms at a lower level or when we have restriction of data confidentiality, or we can't train models in the cloud, custom programming can be done with any of the various frameworks and technologies available on the market.

Available Cloud Solutions:

> Azure ML and Azure Cognitive Service

> Amazon ML and Amazon AI

> Google Cloud Machine Learning and Prediction API

> IBM Watson Services and IBM Watson Machine Learning combined with IBM SPSS Modeler or IBM Data Science Experience

Microsoft frameworks

If you have experience developing with .NET and LINQ, then Accord.NET or NUml frameworks and the library math.net. are interesting options to consider.

Learn more: Furthermore, we recommend using the functional language F# and reading the books *Machine Learning Projects for .NET* developers by M. Brandewinder and *Mastering .NET Machine Learning* by J. Dixon.

Python

Python has many packages that can be added through pip to achieve a customized development kit appropriate for the development of this type of application. If you would like to start prototyping this kind of application, one of the favorite and most recommended options is the scikit-learn package because it has good documentation, an active community and it's easy to use; plus, it provides a collection of algorithms and traditional Machine Learning techniques that one can easily use without needing to implement all the heavy aspects of those algorithms.

Another good option when using Python is the Anaconda distribution, which simplifies the whole installation of the most popular and important packages to work with big-data projects, data-science projects and predictive analysis, such as NumPy , SciPy , Matplotlib, Jupyter and the already mentioned scikit-learn, among other things.

R

The R language is used, in general, by those who come from the math and statistics fields. Many Data Scientists usually use R and feel very comfortable with its language.

Learn more: If you don't have any experience with R and you are curious about it, we recommend the interactive course available in Code School and the open source IDE RStudio to run your first scripts. You can also find many resources about R in the r project website.

An interesting feature about R is that if you use SQL Server as a database engine, you can deploy your models directly into the database engine (from SQL Server 2016 version) and use them in your SQL queries merging scripts from R. This way, you can avoid needing to read and take datasets from the database to the R application. In a similar way, from the CTP 2.0 version of SQL Server 2017, Microsoft extends the support for Machine Learning in the database engine by adding Python as another supported language. That is why Microsoft has changed “R Services” for “Machine Learning Services”; and both R and Python are now two options to use this tool.

Deep Learning Libraries

Tensorflow

Tensorflow was developed by Google and shared with the community as an open source project at the end of 2015. In less than two years, it has gained significant popularity, especially because of the fact that it's used in production environments by some of Google's products, giving it good backing and reputation.

Benefits

Among the main features of TensorFlow, we highlight its capacity to process in gpu cards, which optimizes both the vectorial & matrix operations and the deployment at great scale for distributed solutions.

It comes with APIs for Python, C++ and java (among others) however, the Python API is the most complete and recommended option. The tools included in **TensorBoard** have also been of great help for the visualization of data generated in TensorFlow.

Disadvantages

Regarding its negative aspects, the learning curve is steep because it's a low level library, and both the library and the community around it are starting to mature: the 1.0 version was launched in mid-February of 2017.

Wrappers

At times, it can still be difficult to find documentation or examples to illustrate specific problems we may face. For this reason, it can be useful to evaluate using a wrapper over Tensorflow which provides a higher level of abstraction with the purpose of mitigating

some of these problems. Among them, we highlight:

- *Keras* (it also includes support for Theano)
- *TFlearn*
- *TF-Slim*
- *Skflow* (recommended if you have experience with scikit-learn, originally)

was a separate project, but it was included later on in the Learn module inside *TensorFlow*.

EXPERT´S CORNER

Alternatives to Tensorflow:

- Theano
- Caffe
- Caffe 2 (Facebook)
- Torch
- Microsoft Cognitive Toolkit (before CNTK)
- MxNet

GETTING READY FOR “MACHINE LEARNING FIRST” APPS

SMART APPLICATIONS

When I started working in the software development industry, in the 2000s, there was a great shift from desktop applications to web applications. One could access them outside the office, at any time and from any web browser regardless of the operating system. Wow!

This was a real turning point and it implied that developers had to adapt to the new challenges that were cropping up, such as understanding the concept of distributed applications, stateless protocols such as http and how to simulate stateful applications over it, the new requirements for security or performance, the development concepts for the client or server side, etc.

Then, during 2007, with the announcement of the iPhone and the first iPads in 2010, the tendency was to start creating mobile applications or to somehow adapt the known web to work and look good on those devices. Such was the case that the concept “mobile first” was born in the software development world, and by 2014 mobile web access surpassed desktop. This was another turning point which led experts to develop applications for mobile devices (or at least to work with daily concepts such as responsive design, native or hybrid development), and desktop or web applications.

In a similar way, we now face a new tendency in software development which is the rise of smart applications. Also, because the **abundance of data** is crucial for the **recognition of patterns**, the use of Machine Learning in the information age has been empowe-

red by the Internet of Things (IoT) and big-data; seizing their capacity of generation, storage and processing of a great volume of data in real time at low cost. In some places the idea of **"Machine Learning First"** is already appearing.

We hope we helped you understand the potential that exists in the development of smart applications, the main concepts and definitions that appear when addressing this type of solution, the typical fields of use and the tools and products involved. Also, if you reached this point, we suppose that it's because you are excited about the topic and are thinking about starting to work with it. Are we right? If we are, feel free to reach out and discuss your idea with us. We'll be glad to help!

CONTACT US!

ABOUT THE AUTHOR–

Waldemar is a Computer Engineer with more than 12 years of experience in the software industry. At UruIT, he has developed computer systems for various sectors such as Gamification, Human Resources and Automation Testing. Since 2014, he also teaches as an Auxiliar Professor at the Faculty of Engineering at Universidad de la Republica (FIng - CETP/UTU, UdelaR) giving courses on systems integration, enterprise architectures and .NET technologies. In recent years, he has studied recommender systems and he is currently working on his Computer Science Master's thesis (PEDECIBA) in areas related to operational research, machine learning and natural language processing.

ABOUT URUIT–

UruIT is a boutique software development company with offices in the USA and development centers in Uruguay and Colombia. We specialize in web application development and mobile solutions. Since 2007 we've pioneered the nearshore outsourcing model, assembling teams of senior-only engineers, UX/UI and Product Management roles. Our clients are some of the most innovative VC-backed startups and industry leaders in the US.

DID YOU LIKE IT?

HELP A COLLEAGUE

SHARE THIS EBOOK