* earlier, google used hard code to give page recommendations for queries, but later they developed their deep learning model called Rank Brain.
* if we have labelled data that is data for which you know it is a good answer or a bad answer that data can be used with a machine learning model.

# Housing Rental Recommendation :

* ingest previous ratings that are done by the user.
* Train the model based on user ratings for every house in the database.
* pick top-rated houses for a recommendation, which aren't seen by the user.
* The model is based on user's previous rating to houses and other people's rating to the user unrated houses. (by other people we mean people who show behaviour the same as the user.
* this is not a good model but the very basic one.

## Tech Stack for this system -

* the data that we are dealing with doesn't need to be trained real time we could just train it once a day and store the results, that's why we use batch not stream.
* now, the data can be coming in with a high velocity so for that instead of using a single machine we go for something like apache Hadoop, and to store the recommendations we use an RDBMS like cloud SQL.

let's assume we already have this model running on-premises so we need to migrate it to the cloud.

* we have the model running in apache spark ml on-premises, we will migrate our model to cloud apache-spark ml.
* instead of using an on-prem Hadoop cluster and an SQL DB, we go for cloud data proc and cloud SQL.

When to use what data storage service?

* Cloud storage(PB+): global file system.
* Cloud SQL(GB): RDBMS for transactional and relational data. supports MySQL and Postgres, etc.
* Datastore(TB): transactional no-SQL object-oriented database
* Bigtable(PB): real-time Hightrougput no-SQL append-only data like data from the sensor.
* BigQuery(PB): SQL data warehouse to power data analytics needs.
* Cloud Spanner(TB): if you have a lot of data or multiple databases then use cloud spanner to stored globally.

## problems faced while running an on-prem job-

suppose you have 4 jobs and 2 of the jobs have already used the entire resources they the other 2 will starve out, also if you have only 1 job running at 50% utilization of resources then your other half is powered but not in any use.

hence using cloud data proc we assign different cluster to different jobs

and one can create stop and autoscale the data proc cluster as per user needs.

also if we want to use autoscaling of clusters that are managing cluster according to need then we cannot use HDFS(Hadoop distributed file system) as it has a persistent disk and if the cluster stops the data will be lost or inaccessible. hence we use cloud storage in place of HDFS we can use cloud storage, BigTable or BigQuery instead.

dataproc use autoscaling by using Hadoop's yarn matrix.

**preemptable virtual machines:** Highly-affordable short-lived VM instances. suitable for batch jobs and fault-tolerant workloads. they last only up to 24 hrs. 80% cheaper than in regular instances.

# WHICH MODEL TO USE AND WHEN :

* if there is a ground truth in your data then do **supervised learning**.
* if you want to explore the data for unknown relationships then do **unsupervised learning** and one can start with clustering model.

**supervised learning -**

1. **forecasting** - predicting next months sales figures etc. generally applied on numeric data. Models - linear regression,
2. **classify** - high risk or low risk, buy or no buy etc. generally applied on string data. Models - Logistic Regression.
3. **recommend** - product recommendation. Models - Matrix Factorization.

## *one should set a benchmark before training the model i.e. what threshold of accuracy one expects from the model.*

BigQuery has all this and even more, models built-in and one can just use them without coding that's called AutoML.