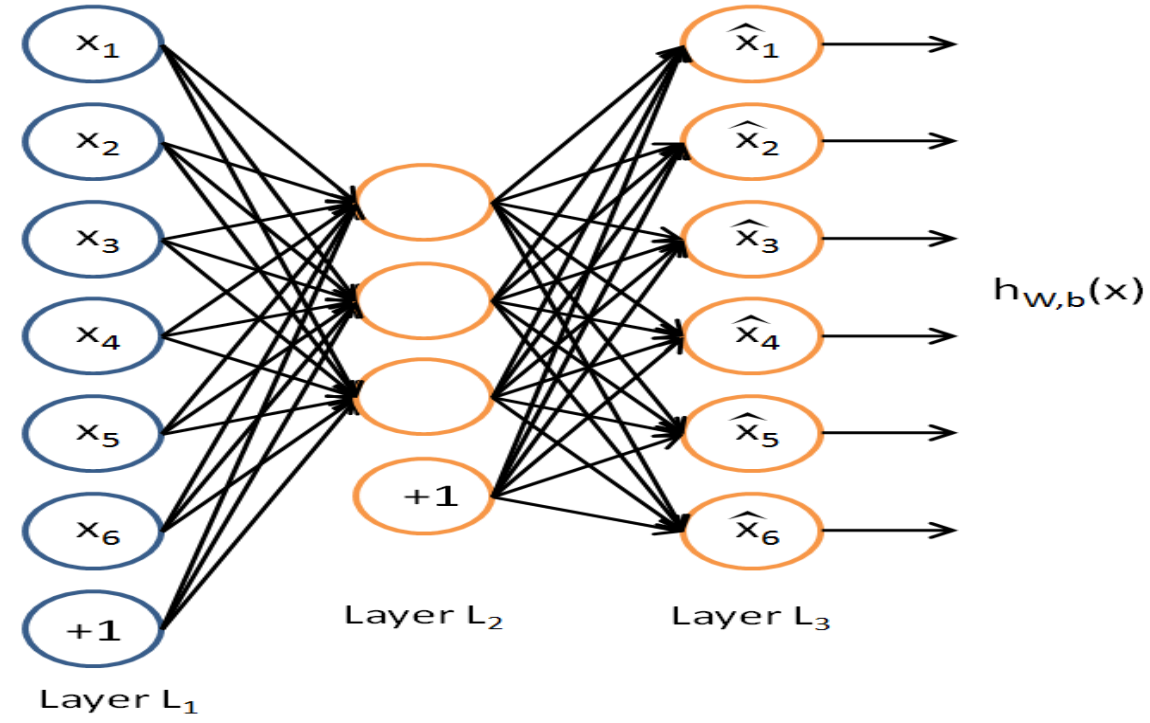
DATA SCIENCE INTERVIEW PREPARATION (30 Days of Interview Preparation)

DAY 15

Q1. What is Autoencoder?

Answer:

Autoencoder neural network: It is an unsupervised Machine learning algorithm that applies backpropagation, setting the target values to be equal to the inputs. It is trained to attempt to copy its input to its output. Internally, it has the hidden layer that describes a code used to represent the input.



It is trying to learn the approximation to the identity function, to output x ̂ x^ that is similar to the xx.

Autoencoders belongs to the neural network family, but they are also closely related to PCA (principal components analysis).

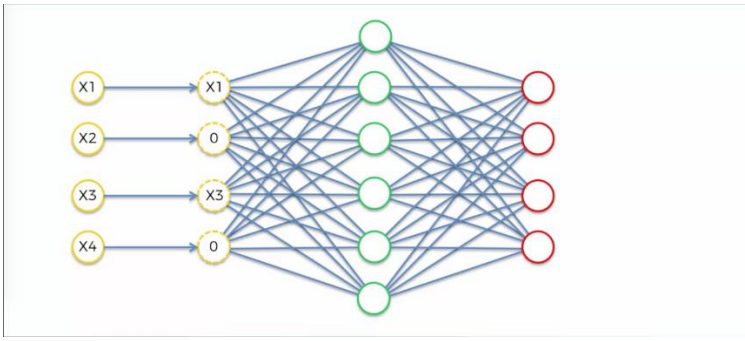
Auto encoders, although it is quite similar to PCA, but its Autoencoders are much more flexible than PCA. Autoencoders can represent both liners and non-linear transformation in encoding, but PCA can perform linear transformation. Autoencoders can be layered to form deep learning network due to its Network representation.

Types of Autoencoders:

1. **Denoising autoencoder**

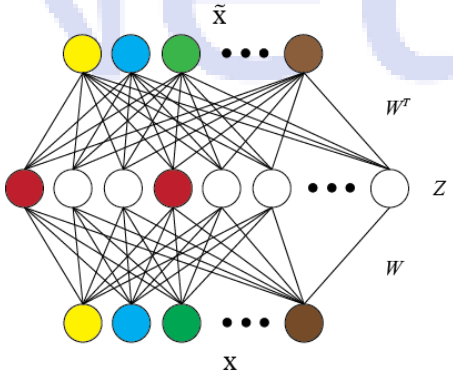
   Autoencoders are Neural Networks which are used for feature selection and extraction. However, when there are more nodes in hidden layer than there are inputs, the Network is risking to learn so-called "Identity Function", also called "Null Function", meaning that output equals the input, marking the Autoencoder useless.

Denoising Autoencoders solve this problem by corrupting the data on purpose by randomly turning some of the input values to zero. In general, the percentage of input nodes which are being set to zero is about 50%. Other sources suggest a lower count, such as 30%. It depends on the amount of data and input nodes you have.



2. **Sparse autoencoder**

An autoencoder takes the input image or vector and learns code dictionary that changes the raw input from one representation to another. Where in sparse autoencoders with a sparsity enforcer that directs a single-layer network to learn code dictionary which in turn minimizes the error in reproducing the input while restricting number of code words for reconstruction. The sparse autoencoder consists a single hidden layer, which is connected to the input vector by a weight matrix forming the encoding step. The hidden layer then outputs to a reconstruction vector, using a tied weight matrix to form the decoder.



Q2. What Is Text Similarity?

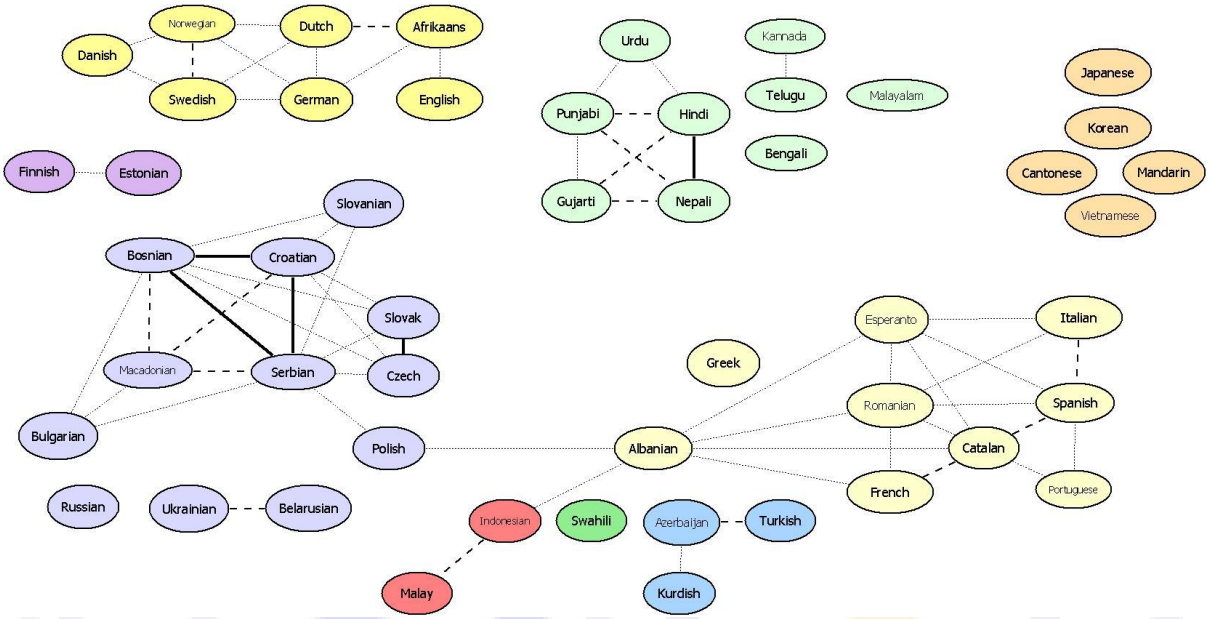
Answer:

When talking about text similarity, different people have a slightly different notion on what text similarity means. In essence, the goal is to compute how 'close' two pieces of text are in (1) meaning or (2) surface closeness. The first is referred to as **semantic similarity**, and the latter is referred to as **lexical similarity**. Although the methods for *lexical similarity* are often used to achieve *semantic similarity* (to a certain extent), achieving true semantic similarity is often much more involved.

Lexical or Word Level Similarity

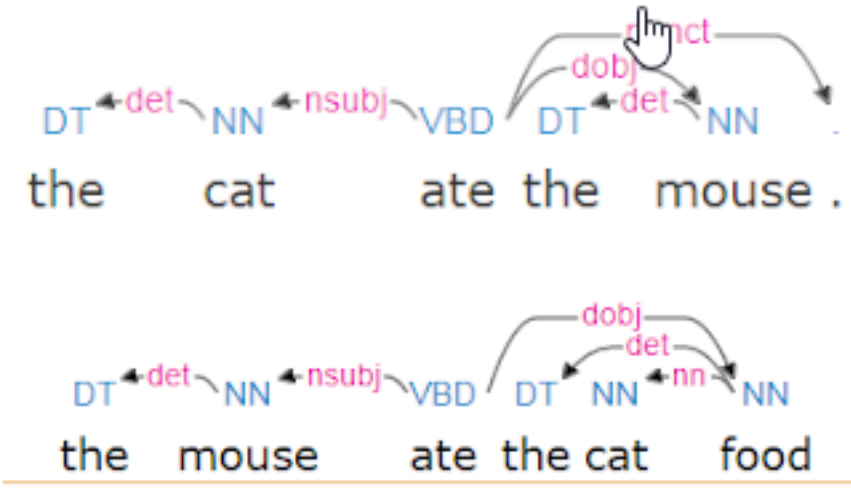
When referring to text similarity, people refer to how similar the two pieces of text are at the surface level. Example- how similar are the phrases *"the cat ate the mouse"* with *"the mouse ate the cat food"* by just looking at the words? On the surface, if you consider only word-level similarity, these two phrases (with determiners disregarded) appear very similar as 3 of the 4 unique words are an exact overlap.

$$Overlap = 'cat\ ate\ mouse' \cap 'mouse\ ate\ cat\ food' = 3$$



Semantic Similarity:

Another notion of similarity mostly explored by NLP research community is how similar in meaning are any two phrases? If we look at the phrases, *" the cat ate the mouse "* and *" the mouse ate the cat food"*. As we know that while the words significantly overlaps, these two phrases have different meaning. Meaning out of the phrases is often the more difficult task as it requires deeper level of analysis.Example, we can actually look at the simple aspects like order of words: *"cat==>ate==>mouse"* and *"mouse==>ate==>cat food"*. Words overlap in this case, the order of the occurrence is different, and we can tell that, these two phrases have different meaning. This is just the one example. Most people use the syntactic parsing to help with the semantic similarity. Let's have a look at the parse trees for these two phrases. What can you get from it?



Q3. What is dropout in neural networks?

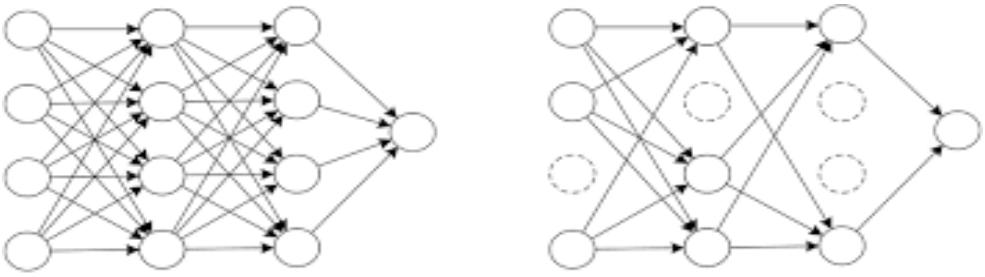
Answer:

When we training our neural network (or model) by updating each of its weights, it might become too dependent on the dataset we are using. Therefore, when this model has to make a prediction or classification, it will not give satisfactory results. This is known as over-fitting. We might understand this problem through a real-world example: If a student of science learns *only* one chapter of a book and then takes a test on the *whole* syllabus, he will probably fail.

To overcome this problem, we use a technique that was introduced by Geoffrey Hinton in 2012. This technique is known as **dropout**.

Dropout refers to ignoring units (i.e., neurons) during the training phase of certain set of neurons, which is chosen at random. By "ignoring", I mean these units are not considered during a particular forward or backward pass.

At each training stage, individual nodes are either dropped out of the net with probability *1-p* or kept with probability *p*, so that a reduced network is left; incoming and outgoing edges to a dropped-out node are also removed.



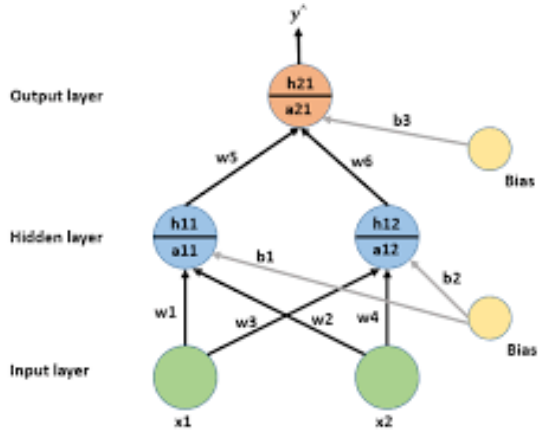
(a) Standard Neural Network

(b) Network after Dropout

Q4. What is Forward Propagation?

Answer:

Input X provides the information that then propagates to hidden units at each layer and then finally produce the output y. The architecture of network entails determining its depth, width, and the activation functions used on each layer. **Depth** is the number of the hidden layers. **Width** is the number of units (nodes) on each hidden layer since we don't control neither input layer nor output layer dimensions. There are quite a few set of activation functions such *Rectified Linear Unit, Sigmoid, Hyperbolic tangent, etc*. Research has proven that deeper networks outperform networks with more hidden units. Therefore, it's always better and won't hurt to train a deeper network.



Q5. What is Text Mining?

Answer:

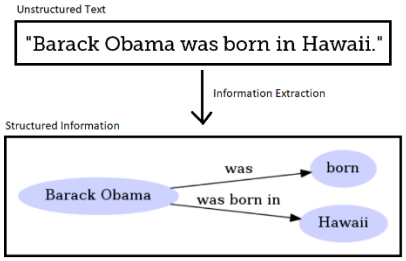
Text mining: It is also referred as ***text data mining***, roughly equivalent to **text analytics**, is the process of deriving high-quality information from text. High-quality information is typically derived through the devising of patterns and trends through means such as statistical pattern learning. Text mining usually involves the process of structuring the input text (usually parsing, along with the addition of some derived linguistic features and the removal of others, and subsequent insertion into a database), deriving patterns within the structured data, and finally evaluation and interpretation of the output. 'High quality' in text mining usually refers to some combination of relevance, novelty, and interest. Typical text mining tasks include text categorization, text clustering, concept/entity extraction, production of granular taxonomies, sentiment analysis, document summarization, and entity relation modeling (*i.e.*, learning relations between named entities).

Q6. What is Information Extraction?

Answer:

Information extraction (IE): It is the task of automatically extracting structured information from the unstructured and/or semi-structured machine-readable documents. In most of the cases, this activity concerns processing human language texts using natural language processing (NLP).

Information extraction depends on named entity recognition (NER), a sub-tool used to find targeted information to extract. NER recognizes entities first as one of several categories, such as location (LOC), persons (PER), or organizations (ORG). Once the information category is recognized, an information extraction utility extracts the named entity's related information and constructs a machine-readable document from it, which algorithms can further process to extract meaning. IE finds meaning by way of other subtasks, including co-reference resolution, relationship extraction, language, and vocabulary analysis, and sometimes audio extraction.



Q7. What is Text Generation?

Answer:

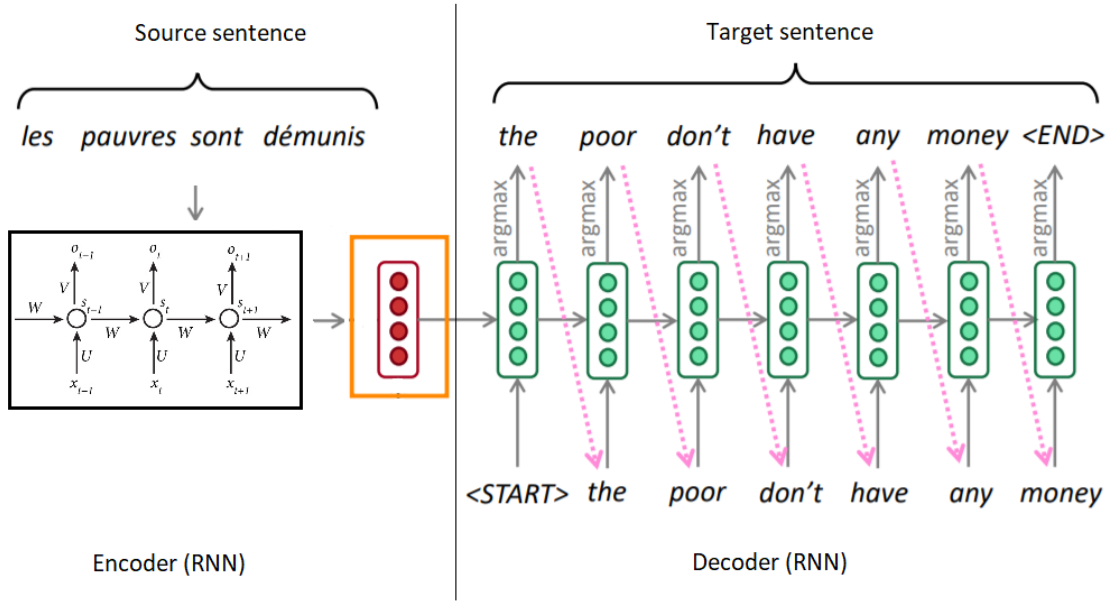
Text Generation: It is a type of the Language Modelling problem. **Language Modelling** is the core problem for several of natural language processing tasks such as speech to text, conversational system, and the text summarization. The trained language model learns the likelihood of occurrence of the word based on the previous sequence of words used in the text. Language models can be operated at the character level, n-gram level, sentence level or even paragraph level.

A language model is at the core of many NLP tasks, and is simply a probability distribution over a sequence of words:

$$p(w_1, \ldots, w_n)$$

It can also be used to estimate the conditional probability of the next word in a sequence:

$$p(w_n \mid w_1, \ldots, w_{n-1})$$



Q8. What is Text Summarization?

Answer:

We all interact with the applications which uses the text summarization. Many of the applications are for the platform which publishes articles on the daily news, entertainment, sports. With our busy schedule, we like to read the summary of those articles before we decide to jump in for reading entire article. Reading a summary helps us to identify the interest area, gives a brief context of the story.

Text summarization is a subdomain of Natural Language Processing (NLP) that deals with extracting summaries from huge chunks of texts. There are two main types of techniques used for text summarization: NLP-based techniques and deep learning-based techniques.

Text summarization: It refers to the technique of shortening long pieces of text. The intention is to create the coherent and fluent summary having only the main points outlined in the document.

How text summarization works:

The two types of summarization, abstractive and the extractive summarization.

1. **Abstractive Summarization:** It select words based on the semantic understanding; even those words did not appear in the source documents. It aims at producing important material in the new way. They interprets and examines the text using advanced natural language techniques to generate the new shorter text that conveys the most critical information from the original text.

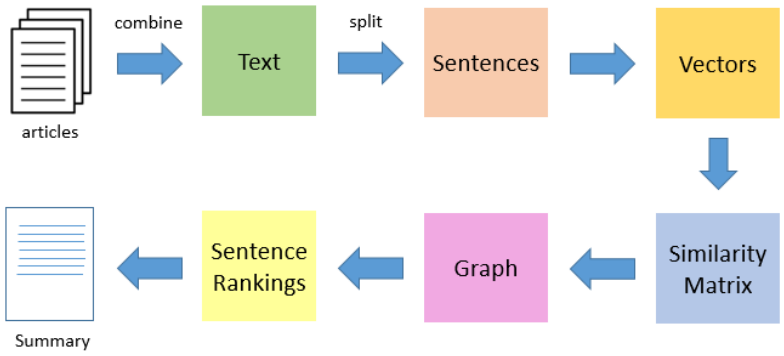
It can be correlated in the way human reads the text article or blog post and then summarizes in their word.

Input document → understand context → semantics → create own summary.

2. **Extractive Summarization:** It attempt to summarize articles by selecting the subset of words that retain the most important points.

   This approach weights the most important part of sentences and uses the same to form the summary. Different algorithm and the techniques are used to define the weights for the sentences and further rank them based on importance and similarity among each other.

   Input document → sentences similarity → weight sentences → select sentences with higher rank.



Q9. What is Topic Modelling?

Answer:

Topic Modelling is the task of using unsupervised learning to extract the main topics (represented as a set of words) that occur in a collection of documents.

Topic modeling, in the context of Natural Language Processing, is described as a method of uncovering hidden structure in a collection of texts.

Dimensionality Reduction:

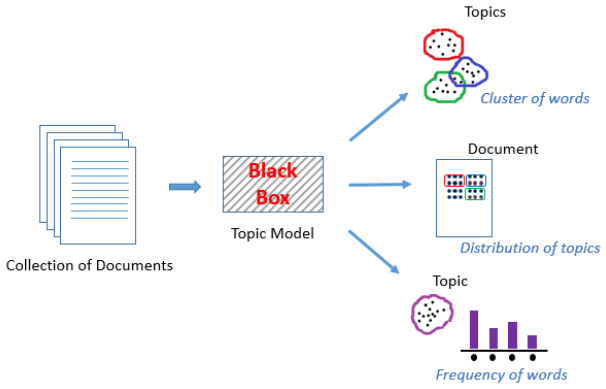
Topic modeling is the form of dimensionality reduction. Rather than representing the text T in its feature space as {Word_i: count(Word_i, T) for Word_i in V}, we can represent the text in its topic space as (Topic_i: weight(Topic_i, T) for Topic_i in Topics).

Unsupervised learning:

Topic modeling can be compared to the clustering. As in the case of clustering, the number of topics, like the number of clusters, is the hyperparameter. By doing the topic modeling, we build clusters of words rather than clusters of texts. A text is thus a mixture of all the topics, each having a certain weight.

A Form of Tagging

If document classification is assigning a single category to a text, topic modeling is assigning multiple tags to a text. A human expert can label the resulting topics with human-readable labels and use different heuristics to convert the weighted topics to a set of tags.



Q10.What is Hidden Markov Models?

Answer:

Hidden Markov Models (HMMs) are the class of probabilistic graphical model that allow us to predict the sequence of unknown (hidden) variables from the set of observed variables. The simple example of an HMM is predicting the weather (hidden variable) based on the type of clothes that someone wears (observed). An HMM can be viewed as the Bayes Net unrolled through time with observations made at the sequence of time steps being used to predict the best sequence of the hidden states.

The below diagram from Wikipedia shows that HMM and its transitions. The scenario is the room that contains urns X1, X2, and X3, each of which contains a known mix of balls, each ball labeled y1, y2, y3, and y4. The sequence of four balls is randomly drawn. In this particular case, the user observes the sequence of balls y1,y2,y3, and y4 and is attempting to discern the hidden state, which is the right sequence of three urns that these four balls were pulled from.

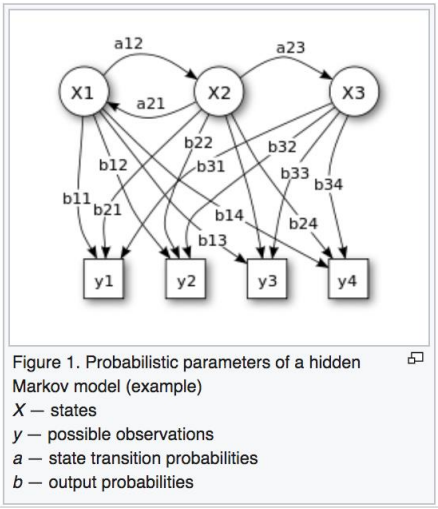
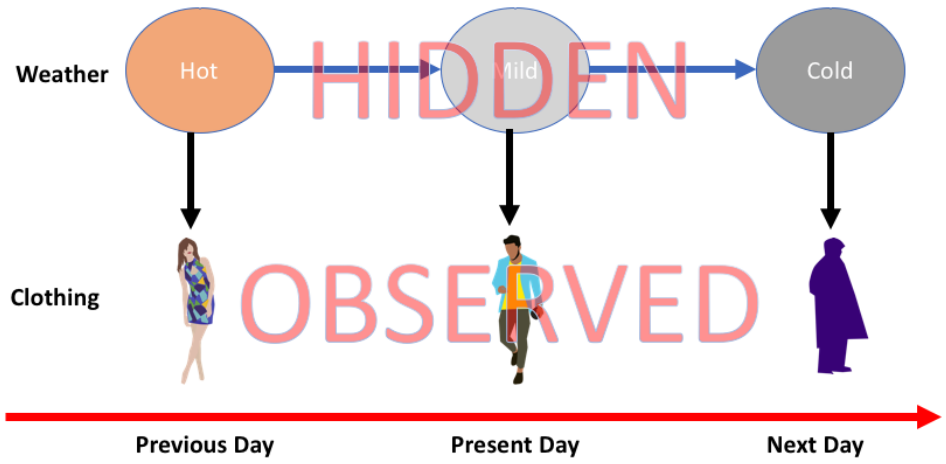


Figure 1. Probabilistic parameters of a hidden Markov model (example)
X — states
y — possible observations
a — state transition probabilities
b — output probabilities

Why Hidden, Markov Model?

The reason it is called the Hidden Markov Model is because we are constructing an inference model based on the assumptions of a Markov process. The Markov process assumption is simply that the "future is independent of the past given the present".

To make this point clear, let us consider the scenario below where the weather, the hidden variable, can be hot, mild or cold, and the observed variables are the type of clothing worn. The arrows represent transitions from a hidden state to another hidden state or from a hidden state to an observed variable.


Weather
Hot
Mild
Cold
HIDDEN
Clothing
OBSERVED
Previous Day
Present Day
Next Day
