Topic Modeling

Agenda

- What is Topic Modeling?
- Examples of Topic Modeling and Topic Classification
- How does Topic Modeling work?
  - Topic Modeling vs Topic Classification
  - Topic Modeling
  - Topic Classification
- Use Cases and Applications:
  - Customer Service
  - Customer Feedback
- Resources:
  - Topic Modeling APIs

Introduction to Topic Modeling

Topic modeling is an unsupervised machine learning technique that's capable of scanning a set of documents, detecting word and phrase patterns within them, and automatically clustering word groups and similar expressions that best characterize a set of documents.

You've probably been hearing a lot about artificial intelligence, along with terms like machine learning and Natural Language Processing (NLP). Especially if you work in a company that processes hundreds, or even thousands of customer interactions every day. Data analysis of social media posts, emails, chats, open-ended survey responses, and more, is not an easy task, and less so when delegated to humans alone.

Introduction to Topic Modeling

That's why many are excited about the implications artificial intelligence could have on their day-to-day tasks, as well as on businesses as a whole. AI-powered text analysis uses a wide variety of methods or algorithms to process language naturally, one of which is topic analysis – used to automatically detect topics from texts.

By using topic analysis models, businesses are able to offload simple tasks onto machines instead of overloading employees with too much data. Just imagine the time your team could save and spend on more important tasks, if a machine was able to sort through endless lists of customer surveys or support tickets every morning.

Introduction to Topic Modeling

In this guide, we're going to take a look at two types of topic analysis techniques: topic modeling and topic classification. Topic modeling is an 'unsupervised' machine learning technique, in other words, one that doesn't require training. Topic classification is a 'supervised' machine learning technique, one that needs training before being able to automatically analyze texts.

First, we'll delve into what topic modeling is, how it works, and how it compares to topic classification. Then, we'll present various use cases and tools that you can use to easily get started with topic analysis, as well as a series of tutorials that will help you create your own models.

Introduction to Topic Modeling

An example of topic modeling

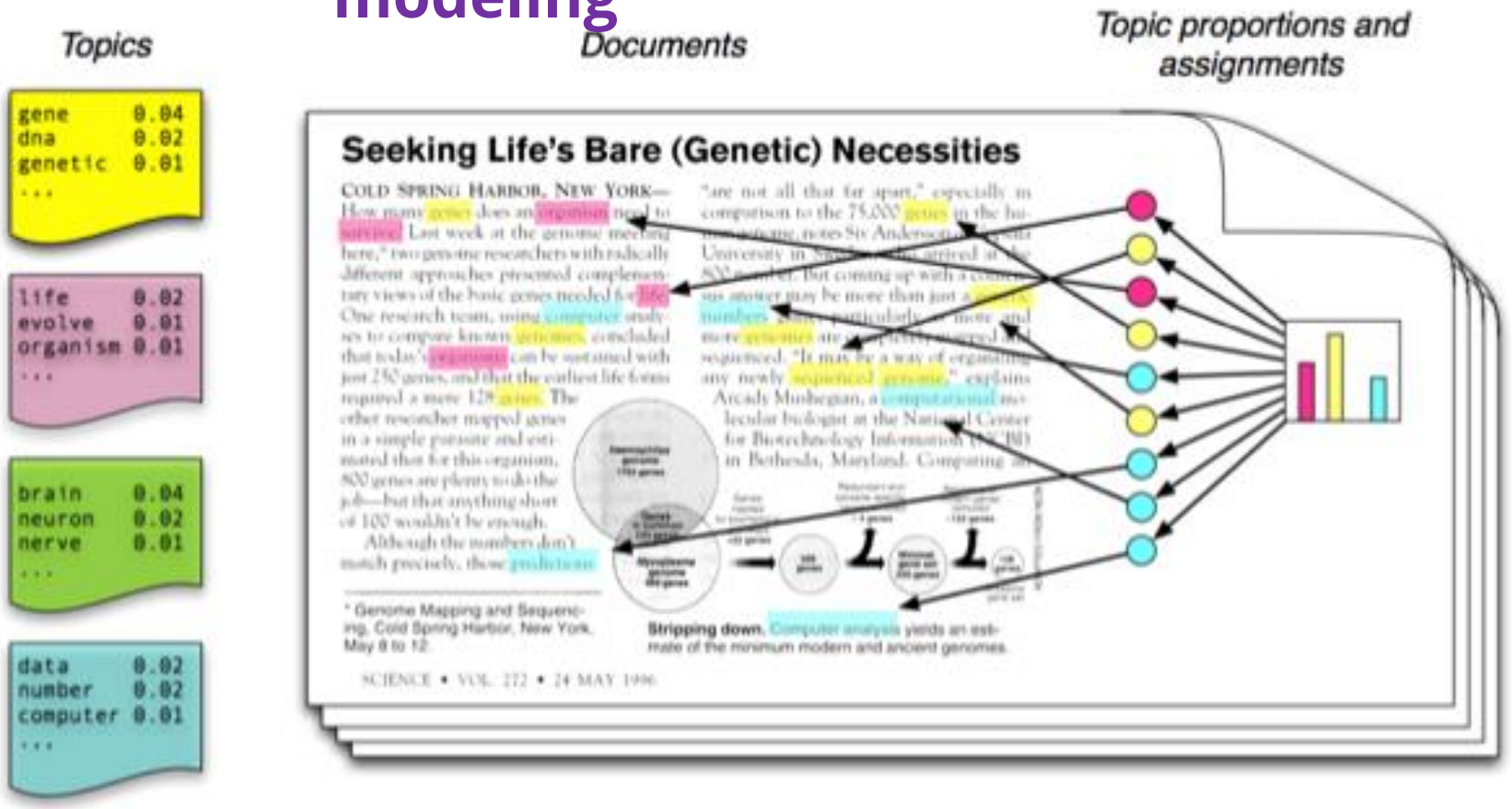


Figure source: Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, *55*(4), 77-84.

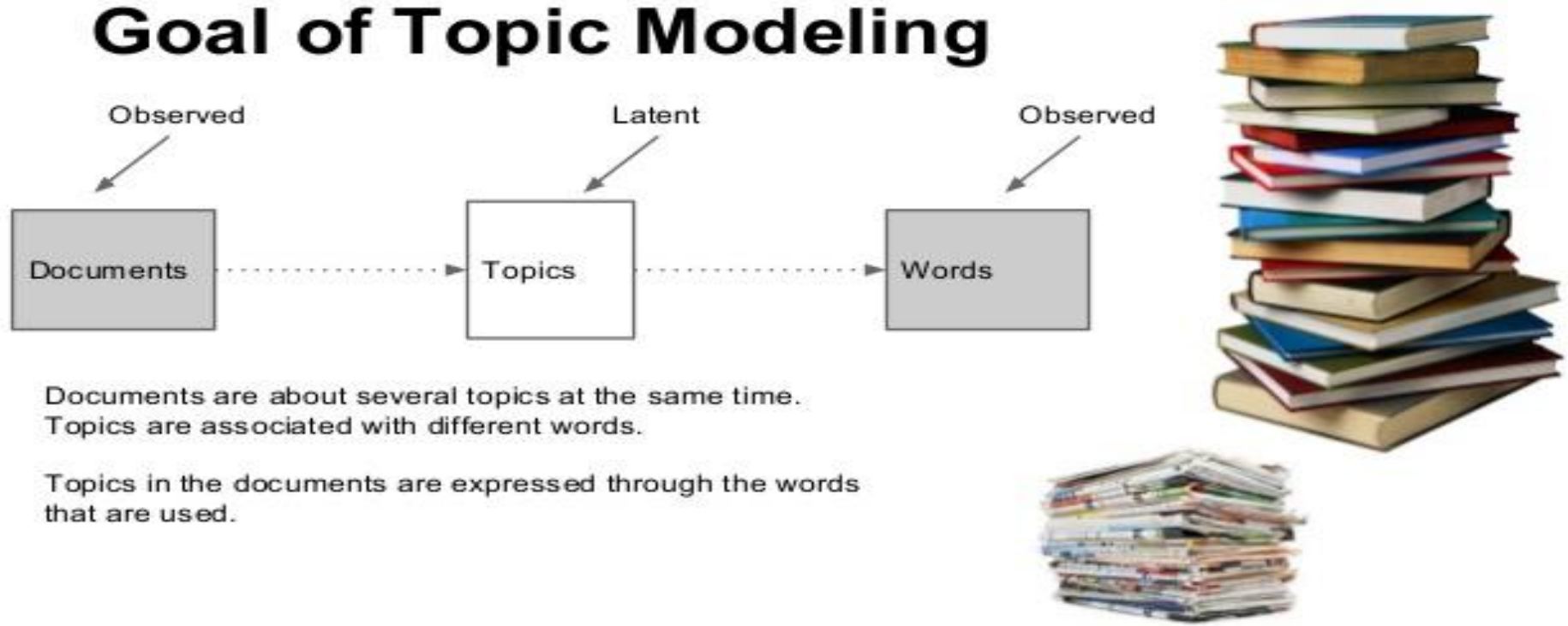
WHAT IS A TOPIC MODEL?

A type of statistical model for discovering the abstract "topics" that occur in a collection of documents.

Topic modeling is a machine learning technique that automatically analyzes text data to determine cluster words for a set of documents. This is known as 'unsupervised' machine learning because it doesn't require a predefined list of tags or training data that's been previously classified by humans.

WHAT IS A TOPIC MODEL?

Since topic modeling doesn't require training, it's a quick and easy way to start analyzing your data. However, you can't guarantee you'll receive accurate results, which is why many businesses opt to invest time training a topic classification model.



WHAT IS A TOPIC MODEL?

Since topic classification models require training, they're known as 'supervised' machine learning techniques. What does that mean? Well, as opposed to text modeling, topic classification needs to know the topics of a set of texts before analyzing them. Using these topics, data is tagged manually so that a topic classifier can learn and later make predictions by itself.

Examples of Topic Modeling and Topic Classification

Topic modeling could be used to identify the topics of a set of customer reviews by detecting patterns and recurring words. Let's take a look at how an 'unsupervised' technique would group the below review for Eventbrite, for example:

"The nice thing about Eventbrite is that it's free to use as long as you're not charging for the event. There is a fee if you are charging for the event – 2.5% plus a $0.99 transaction fee."

By identifying words and expressions such as free to use, fee, charging, 2.5% plus 99 cents transaction fee, topic modeling can group this review with other reviews that talk about similar things (these may or may not be about pricing).

Examples of Topic Modeling and Topic Classification

A topic classification model could also be used to determine what customers are talking about in customer reviews, open-ended survey responses, and on social media, to name just a few. However, these supervised techniques use a different approach. Rather than inferring what similarity cluster the review belongs to, classification models are able to automatically label a review with predefined topic tags. Take this review about SurveyMonkey, for example:

"We have the gold level plan and use it for everything, love the features! It is one of the best bang for buck possible."

Examples of Topic Modeling and Topic Classification

A topic classification model that's been trained to understand these expressions (gold level plan, love the features, and best bang for buck) would be able to tag this review as topics Features and Price.

In short, topic modeling algorithms churn out collections of expressions and words that it thinks are related, leaving you to figure out what these relations mean, while topic classification delivers neatly packaged topics, with labels such as Price, and Features, eliminating any guesswork.

How Does Topic Modeling Work?

It's simple, really. Topic modeling involves counting words and grouping similar word patterns to infer topics within unstructured data. Let's say you're a software company and you want to know what customers are saying about particular features of your product. Instead of spending hours going through heaps of feedback, in an attempt to deduce which texts are talking about your topics of interest, you could analyze them with a topic modeling algorithm.

How Does Topic Modeling Work?

By detecting patterns such as word frequency and distance between words, a topic model clusters feedback that is similar, and words and expressions that appear most often. With this information, you can quickly deduce what each set of texts are talking about. Remember, this approach is 'unsupervised' meaning that no training is required.

Now, let's say you train a model to detect specific topics. That's a whole different kettle of fish, and a step that's needed for topic classification algorithms – a supervised technique. Let's compare the two topic analysis algorithms to further understand the differences between them.

Topic Modeling vs Topic Classification

Topic modeling and topic classification do have one thing in common. They're the most commonly used topic analysis techniques. Apart from that, they're both very different and the one you choose, well, that depends on several factors.

In theory, unsupervised machine learning algorithms such as topic modeling require less manual input than supervised algorithms. That's because they don't need to be trained by humans with manually tagged data. However, they do need high-quality data, and not only that – they need it in bucket loads, which may not always be easy to come by.

Topic Modeling vs Topic Classification

At the end of your topic modeling analysis, you'll receive collections of documents that the algorithm has grouped together, as well as clusters of words and expressions that it used to infer these relations.

Supervised machine learning algorithms, on the other hand, deliver neatly packaged results with topic labels such as Price and UX. Yes, they take longer to set up since you'll need to train them by tagging datasets with a predefined list of topics. But, if you label your texts accurately and refine your criteria, you'll be rewarded with a model that can accurately classify unseen texts according to their topics, as well as results that you can put to use.

Topic Modeling vs Topic Classification

At the end of the day, it comes down to this. If you don't have a lot of time to analyze texts, or you're not looking for a fine-grained analysis and just want to figure out what topics a bunch of texts are talking about, you'll probably be happy with a topic modeling algorithm.

Topic Modeling vs Topic Classification

However, if you have a list of predefined topics for a set of texts and want to label them automatically without having to read each one, as well as gain accurate insights, you're better off using a topic classification algorithm.

Now that we've explained the differences between topic modeling and topic classification, we're going to go into more detail about how each of these machine learning algorithms works... and, yes, things are about to get a bit more technical.

Topic Modeling

Topic Modeling refers to the process of dividing a corpus of documents in two:

- **A list of the topics covered by the documents in the corpus.**
- **Several sets of documents from the corpus grouped by the topics they cover.**

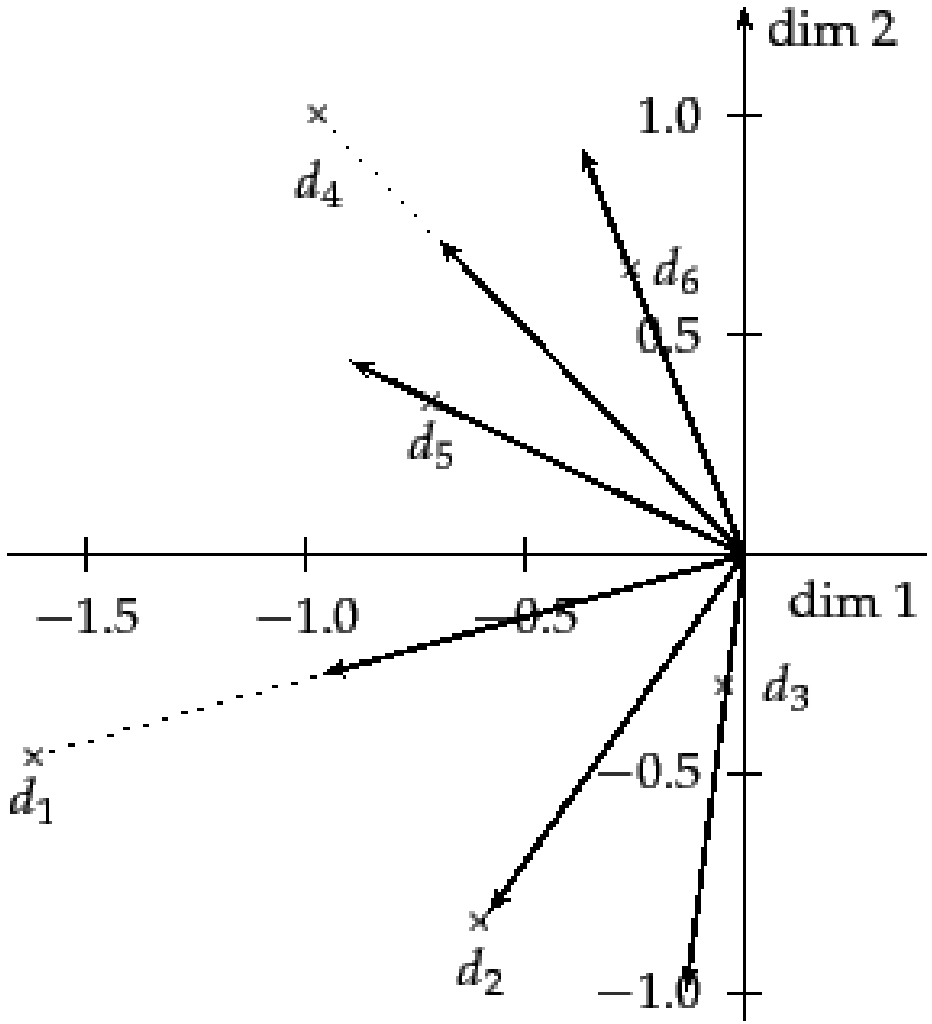
Topic Modeling

The underlying assumption is that every document comprises a statistical mixture of topics, i.e., a statistical distribution of topics that can be obtained by "adding up" all of the distributions for all the topics covered. What topic modeling methods do is try to figure out which topics are present in the documents of the corpus and how strong that presence is.

In this section, we'll help you see the big picture of two topic modeling methods, namely, Latent Semantic Analysis (LSA) _and _Latent Dirichlet Allocation (LDA).

Latent Semantic Analysis (LSA):

Latent Semantic Analysis (LSA) is a mathematical method that tries to bring out latent relationships within a collection of documents on to a lower dimensional space. LSA assumes that words that are close in meaning will occur in similar pieces of text (the distributional hypothesis).



Latent Semantic Analysis (LSA):

A matrix containing word counts per paragraph (rows represent unique words and columns represent each paragraph) is constructed from a large piece of text and a mathematical technique called singular value decomposition (SVD) is used to reduce the number of rows while preserving the similarity structure among columns. Rather than looking at each document isolated from the others it looks at all the documents as a whole and the terms within them to identify relationships.

Latent Semantic Analysis (LSA):

Singular value decomposition can be used to solve the low-rank matrix approximation problem. We then derive from it an application to approximating term-document matrices. We invoke the following three-step procedure to t

- Given C, construct its SVD in the form $C = U\Sigma V^T$.
- Derive from Σ the matrix Σ_k formed by replacing by zeros the $r - k$ smallest singular values on the diagonal of Σ.
- Compute and output $C_k = U\Sigma_k V^T$ as the rank-k approximation to C.;

Where C is the term-document matrix and U, Σ and V^T are SVD computed matrices.

Latent Dirichlet Allocation (LDA)

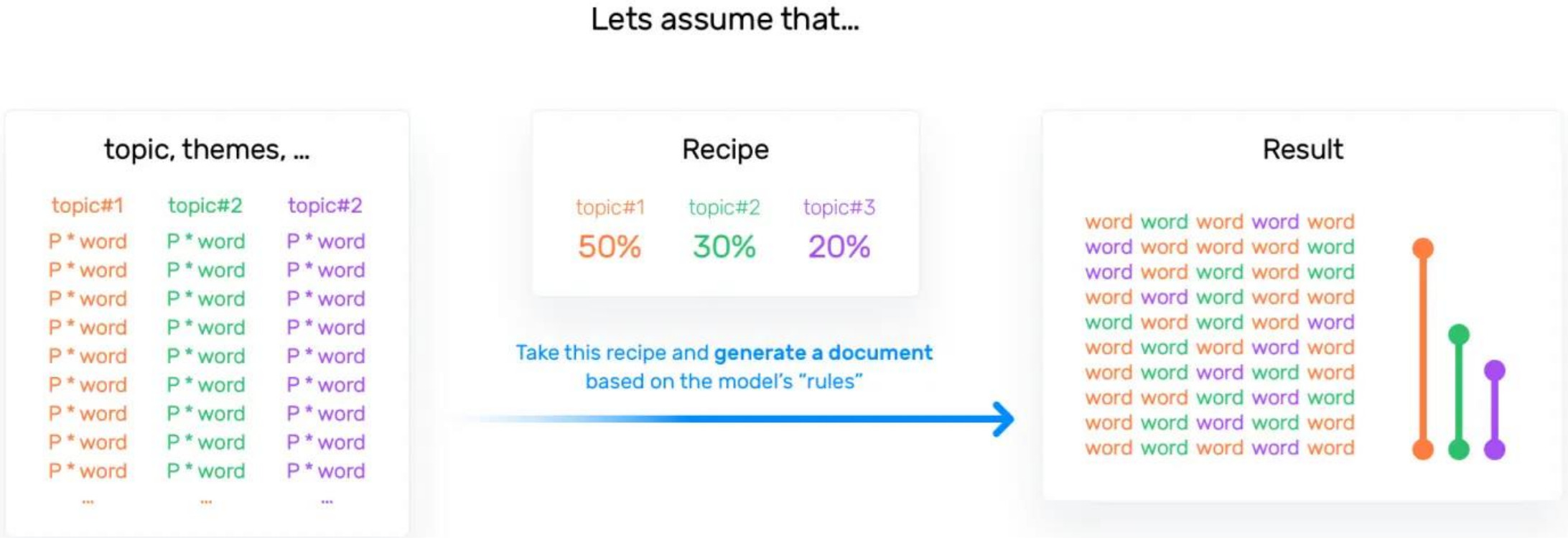
Latent Dirichlet Allocation (LDA) and LSA are based on the same underlying assumptions: the distributional hypothesis, (i.e., similar topics make use of similar words) and the statistical mixture hypothesis (i.e., documents talk about several topics) for which a statistical distribution can be determined. The purpose of LDA is mapping each document in our corpus to a set of topics which covers a good deal of the words in the document.

Latent Dirichlet Allocation (LDA)

Is a generative statistical model that allows sets of observations to be explained by unobserved groups that explain why some parts of the data are similar. For example, if observations are words collected into documents, it posits that each document is a mixture of a small number of topics and that each word's presence is attributable to one of the document's topics.

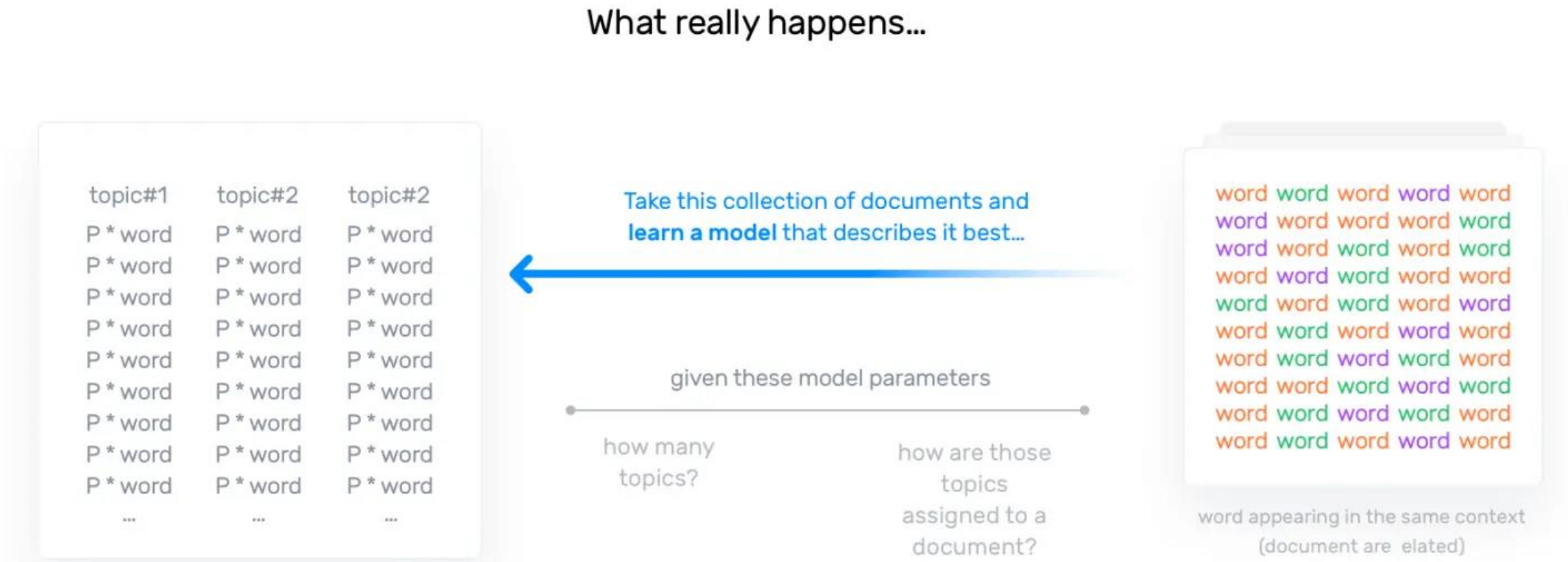
Latent Dirichlet Allocation (LDA)

LDA assumes that topics and documents look like this:



Latent Dirichlet Allocation (LDA)

And, when LDA models a new document, it works this way:



Latent Dirichlet Allocation (LDA)

The main difference between LSA and LDA is that LDA assumes that the distribution of topics in a document and the distribution of words in topics are Dirichlet distributions. LSA does not assume any distribution and therefore, leads to more opaque vector representations of topics and documents.

Topic Classification

If you're running a topic classification analysis, you'll need to predefine a list of topics. For example, if you're a software company and you have a set of customer reviews you want to analyze, you'll probably include topics such as Functionality , Usability, and Reliability on your list.

To understand the ins and outs of this supervised machine learning model, There are three ways you can approach automated topic classification: rule-based systems, machine learning systems, and hybrid systems.

Use Cases & Applications:

From sales and marketing to customer support and product teams, topic modeling and topic classification can help eliminate manual and repetitive tasks, as well as speed up processes in a simple and cost-effective way.

To understand how machine learning could help you, let's take a closer look at the areas in which topic classification and topic modeling are making waves:

- Customer Service
- Customer Feedback

Customer Service:

In this day and age, customer service can break or make a company. It's no longer about having an awesome product when there are competitors out there with similar products. Your unique selling point (USP) now needs to be about constantly delivering a kick-ass customer service that will make you stand out from the crowd.

Customer Service:

Topic modeling and topic classification models can be used in customer support to help teams handle large amounts of data by:

Automatically tagging customer support tickets according to topic or recognizing patterns and delivering results in the form of frequently occurring words and expressions.

Customer Service:

Automatically triaging and routing conversations to the most appropriate team. For example, tickets tagged Billing Issues or Refunds, or containing expressions such as ‘credit card transaction’, ‘subscription error’, and so on, would be sent to the accounts department. Likewise, queries tagged with Bug Issues and Software, or containing expressions such as ‘strange glitch’ and ‘app isn’t working’ would be sent to the dev team.

Customer Service:

Automatically detecting the urgency of a support ticket and prioritizing accordingly. For example, if a ticket is tagged as Bug Issue, Urgent, or a machine recognizes expressions such as ‘right away’, ‘immediate attention’ etc. This approach to text analysis has helped companies avoid a potential PR crisis, and even make the most out of a bad situation.

Getting insights from **customer support conversations**

Customer Feedback:

We've all come across the term customer-centric – a strategy that's based on putting your customer first, and at the core of your business. That means listening to the Voice of Customer (VoC), in other words, what customers have to say, via reviews, social media posts, emails, chats and surveys, and responding in a way that will make them want to use your service or product again, and even recommend it to others. After having a positive experience with a company, 77% of customers would recommend it to a friend.

The best way to approach this customer-centric strategy? With machine learning in tow.

Customer Feedback:

The best way to approach this customer-centric strategy? With machine learning in tow.

Not only can you use topic classification and topic modeling for processing customer feedback and responding in a more timely and effective manner, you can also use it to make more informed decisions, either on the spot when dealing with individual customers, or when making improvements to your product or service.

Resources:

We'll introduce you to a selection of tools, including topic classification and topic modeling APIs, open-source libraries and SaaS APIs, to steer you in the right direction. If you want to dive deeper into how topic detection works, and practice what you've learned, our recommendations for papers and online courses are sure to pique your interest. Finally, we'll provide you with some tutorials that will help you create your own topic classification and topic modeling tools.

Topic modeling APIs:

Application programming interfaces (APIs) are a great way to seamlessly connect applications and extend the functionality of your apps. Luckily, there are plenty of topic modeling tools with their own API, and various languages in the data science community that are ideal for these machine learning models. Let’s take a closer look:

Topic modeling APIs:

Open source:

If you know how to code, there are many open-source libraries for implementing a topic modeling solution from scratch. These are great because they offer flexibility and customization and give you complete control of the whole process – from the pre-processing of data (tokenization, stop words removal, stemming, lemmatization, etc.,), to feature extraction and training of the model (choosing the algorithm and its parameters).

Topic modeling APIs:

Python:

1. **Python** is one of the most popular programming languages for machine learning and data analysis. It implement NLP models.
2. **NLTK** is a framework that is widely used for topic modeling and text classification.
3. **SpaCy** is the fastest framework for training NLP models.
4. Scikit-learn provides a wide variety of algorithms for building machine learning models.

Topic modeling APIs:

SaaS APIs: Here are some machine learning services that you can try out for free:

- MonkeyLearn
- Amazon Comprehend
- IBM Watson
- Google Cloud NLP
- Aylien
- MeaningCloud
- BigML