Unsupervised Text Classification with Topic Models and Good Old Human Reasoning

Use your brain and your data interpretation skills, and create production-ready pipelines without labeled data

Márton Kardos · Follow

9 min read · Aug 3

One challenge I keep on encountering in my daily work is labeling textual data without access to gold-standard labels. This is by no means a trivial task and in this article I set out to show you a way to do this with relative accuracy while keeping your pipeline interpretable and easy to tweak.

Some readers might already be flirting with the idea of using transformer-based zero shot models for this task (or LLMs) but here I will present a couple of reasons for why this might be problematic:

—Zero-shot learning is a black box. Understanding the implications of your prompt/class label choices is rather hard, and you rarely ever have a good intuition of what all of your seemingly arbitrary choices will affect.
— Transformers and LLMs are slow and costly. Using OpenAI's API costs crazy amounts of money, and it might be impractical due to it being rather

slow. You can of course self-host a smaller transformer model, but it will still require a lot of computational resources if you want things to be snappy and responsive (which is usually a requirement in production).

In this case I would say topic models are a very reasonable compromise. They might not be nearly as smart as zero-shot transformer models, and you will have to do a lot more manual labor to employ them in practice, but they give you more fine-grained control over the process, and give much more interpretable results, not to speak of the performance benefits.

The Workflow

In this article I will walk you through a workflow for creating machine learning pipelines to label novel texts using topic models and good old cold hard algorithmic rules.

The Data

For demonstrating my point I will be cheating a bit and I will use a labelled dataset simply to demonstrate the effectiveness of the approach I'm proposing. I will only use the labels for evaluation though, the whole process of creating a pipeline will be based on unsupervised learning and our own human intuition. This dataset is 20newsgroups, that you can easily load with scikit-learn.

```
pip install scikit-learn
```

```
from sklearn.datasets import fetch_20newsgroups
import numpy as np

newsgroups = fetch_20newsgroups(subset="all")
```

```
corpus = newsgroups.data

# Sklearn gives the labels back as integers, we have to map them back to
# the actual textual label.
group_labels = [newsgroups.target_names[label] for label in newsgroups.target]

print(np.unique(group_labels))
--------------------------------------------------------------------
array(['alt.atheism', 'comp.graphics', 'comp.os.ms-windows.misc',
       'comp.sys.ibm.pc.hardware', 'comp.sys.mac.hardware',
       'comp.windows.x', 'misc.forsale', 'rec.autos', 'rec.motorcycles',
       'rec.sport.baseball', 'rec.sport.hockey', 'sci.crypt',
       'sci.electronics', 'sci.med', 'sci.space',
       'soc.religion.christian', 'talk.politics.guns',
       'talk.politics.mideast', 'talk.politics.misc',
       'talk.religion.misc'], dtype='<U24')
```

Let's say that I would like to label texts according to whether they are space related or not, so that we can compare our unsupervised classification performance to the actual labels. I will even go as far as to split the data into train and test sets, so that we can make sure that we are not informed on the information that the model is getting tested on (aka the topic model will not be trained on the test set).

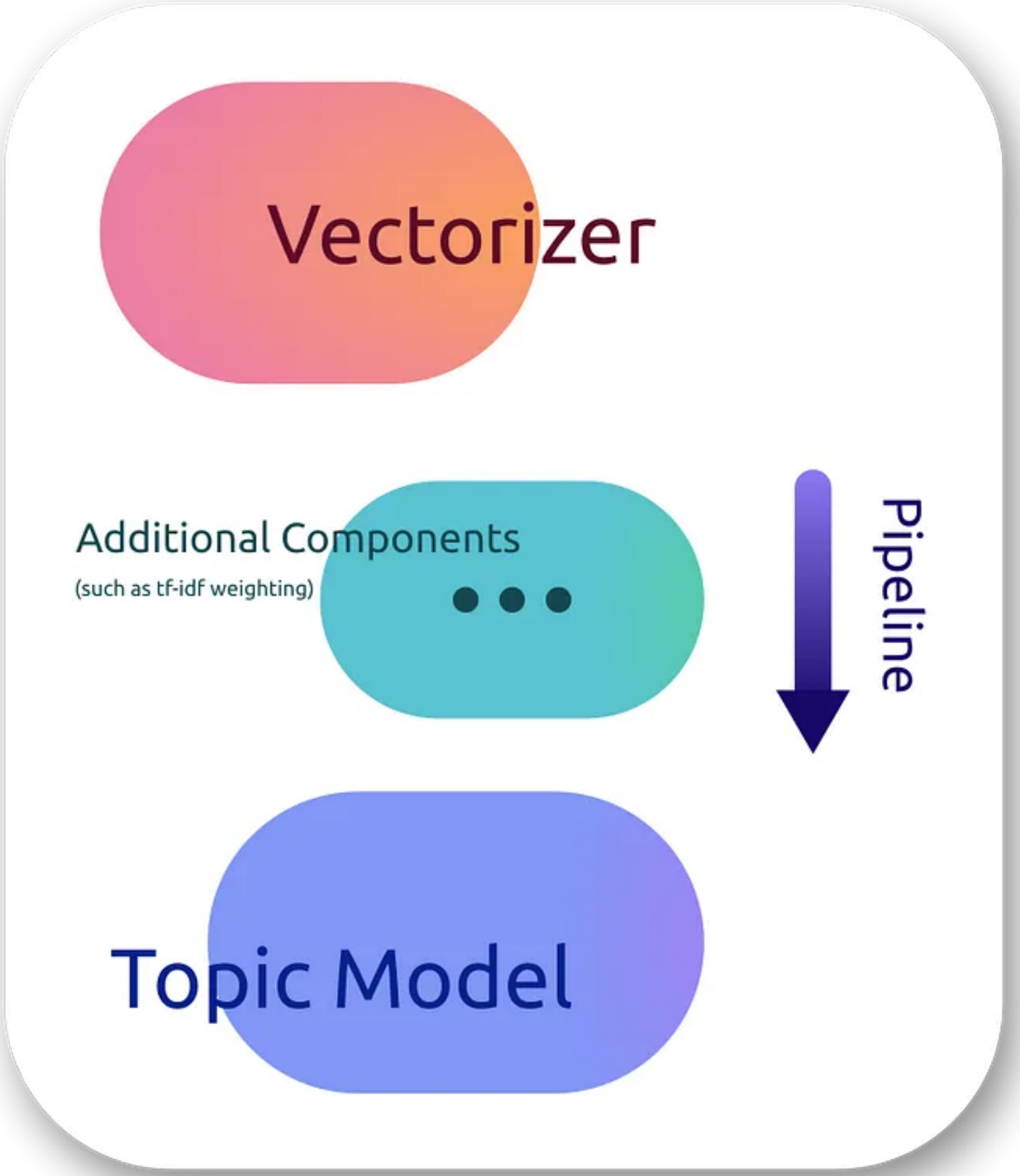
```
from sklearn.model_selection import train_test_split
is_about_space = np.array(group_labels) == "sci.space"

X_train, X_test, y_train, y_test = train_test_split(corpus, is_about_space)
```

The Unsupervised Model

We can use topicwizard for creating easy-to-use topic pipelines and then interpret the topics.

```
pip install topicwizard
```

Creating a topic pipeline in topicwizard is thought of as chaining a vectorizer and a decomposition model together.



As a vectorizer we will be using scikit-learn's built in CountVectorizer and set some sensible looking default frequency cutoffs and will filter English stop words.

We will be using Non-negative Matrix Factorization as our topic model as it is very fast (both training and inference) and it usually works reasonably well. I will settle on 30 topics which is a completely arbitrary number.

```
from sklearn.decomposition import NMF
from sklearn.feature_extraction.text import CountVectorizer
from topicwizard.pipeline import make_topic_pipeline

# Setting up topic modelling pipeline
vectorizer = CountVectorizer(max_df=0.5, min_df=10, stop_words="english")
# NMF topic model with 20 topics
nmf = NMF(n_components=30)

topic_pipeline = make_topic_pipeline(vectorizer, nmf)
topic_pipeline.fit(X_train)
```

Model Interpretation

topicwizard comes with a lot of built-in visualizations to interpret topic models. Right now our main interest is which topics might contain most of the space-related words.

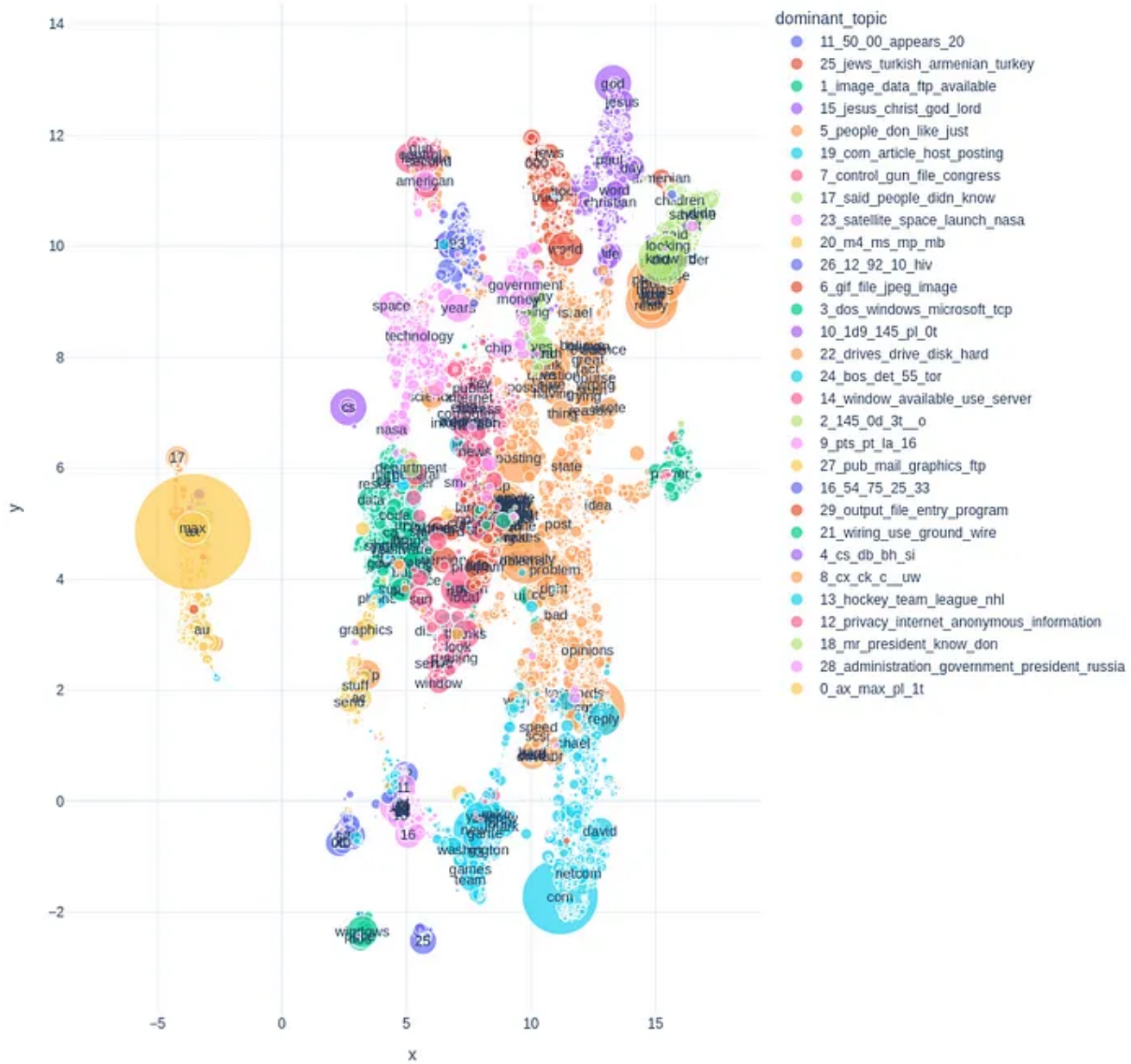
For this we will be using topicwizard's figure API. First let's have a look at the most relevant words for each topic by creating a set of bar charts.

```
from topicwizard.figures import topic_barcharts

topic_barcharts(X_train, pipeline=topic_pipeline, top_n=5)
```

Most of it is unfortunately junk, we probably should have done a better job of cleaning the data set, but *23_satellite_space_launch_nasa* looks rather promising.

Let's have a look at a map of the words in the topic model to get a feel for where they are located in relation to each other.

```
from topicwizard.figures import word_map

word_map(X_train, pipeline=topic_pipeline, top_n=5)
```



We can see that there is a group of words that are very space-y and they are located quite close to words about certain tech-related words too.

We can also check which topics the words “space” and “astro” belong to along with let’s say 20 of their closest associations. We will only show the top 8 topics.

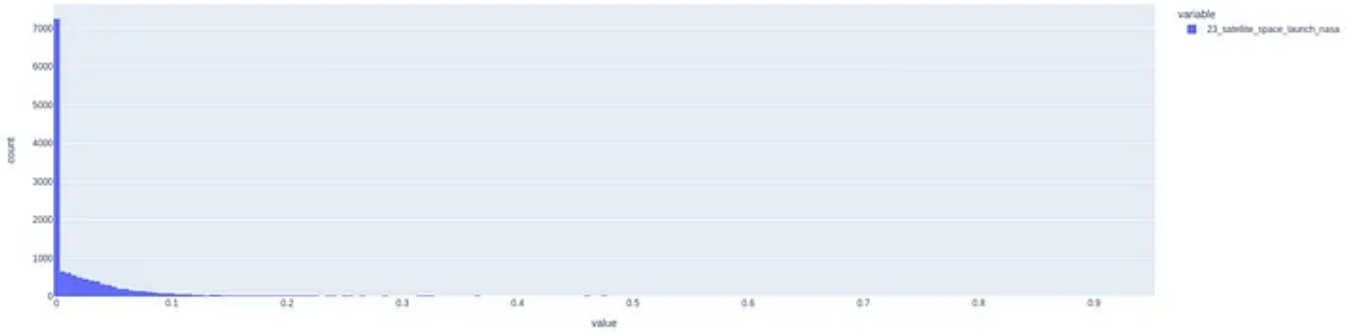
```
from topicwizard.figures import word_association_barchart

fig = word_association_barchart(
  ["space", "astro"],
```

```
    corpus=X_train,
    pipeline=topic_pipeline,
    n_association=20,
    top_n=8
)
```

We can see that by far the most dominant topic is the one we already identified, I will take the freedom for myself to only attend to this one in our analyses from now on.

Let's transform our training corpus and look at the distribution of importances for this topic, so that we can select a reasonable threshold. First I will set the output of the topic model to be a data frame, then we can see the distribution with a plotly express histogram.

```
import plotly.express as px

topic_pipeline.set_output(transform="pandas")

topic_df = topic_pipeline.transform(X_train)
px.histogram(topic_df["23_satellite_space_launch_nasa"])
```



We can see that the vast majority of texts are under 0.1. I say we try setting a threshold at 0.05 and see a random sample of texts we get out of this.

```
topic_df["content"] = X_train
sample = topic_df[topic_df["23_satellite_space_launch_nasa"] > 0.05].content.sam
for text in sample:
  print(text[:200])
```

```
From: CPKJP@vm.cc.latech.edu (Kevin Parker)
Subject: Insurance Rates on Performance Cars SUMMARY
Organization: Louisiana Tech University
Lines: 244
NNTP-Posting-Host: vm.cc.latech.edu
X-Newsreader: NN
From: pjs@euclid.JPL.NASA.GOV (Peter J. Scott)
Subject: Re: Did Microsoft buy Xhibition??
Organization: Jet Propulsion Laboratory, NASA/Caltech
```

```
Lines: 8
Distribution: world
Reply-To: pjs@euclid.jpl.na
From: ml@chiron.astro.uu.se (Mats Lindgren)
Subject: Re: Comet in Temporary Orbit Around Jupiter?
Organization: Uppsala University
Lines: 14
Distribution: world
NNTP-Posting-Host: chiron.astro.uu.se


From: mike@gordian.com (Michael A. Thomas)
Subject: Re: The Role of the National News Media in Inflaming Passions
Organization: Gordian; Costa Mesa, CA
Distribution: ca
Lines: 13

In article <1qjtmjIN
From: leech@cs.unc.edu (Jon Leech)
Subject: Space FAQ 04/15 - Calculations
Supersedes: <math_730956451@cs.unc.edu>
Organization: University of North Carolina, Chapel Hill
Lines: 334
Distribution: worl
From: wls@calvin.usc.edu (Bill Scheding)
Subject: Re: "Full page" PB screen
Organization: University of Southern California, Los Angeles, CA
Lines: 14
Distribution: world
NNTP-Posting-Host: calvin.usc
From: ghelf@violet.berkeley.edu (;;;;RD48)
Subject: Re: Soyuz and Shuttle Comparisons
Organization: University of California, Berkeley
Lines: 11
NNTP-Posting-Host: violet.berkeley.edu

Are you guys ta
From: gsh7w@fermi.clas.Virginia.EDU (Greg Hennessy)
Subject: Re: Keeping Spacecraft on after Funding Cuts.
Organization: University of Virginia
Lines: 13

In article <1r6aqr$dnv@access.digex.net> prb@
From: oeth6050@iscsvax.uni.edu
Subject: ****COMIC BOOK SALE****
Organization: University of Northern Iowa
Lines: 36

Hello,
        my name is John and I have the following comic books for sale - plea
```

```
From: shafer@rigel.dfrf.nasa.gov (Mary Shafer)
Subject: Re: Inner Ear Problems from Too Much Flying?
Article-I.D.: rigel.SHAFER.93Apr6095951
Organization: NASA Dryden, Edwards, Cal.
Lines: 33
In-Reply
```

Hmm some of these texts do not seem to have much to do with space, let's set a higher threshold.

```
topic_df["content"] = X_train
sample = topic_df[topic_df["23_satellite_space_launch_nasa"] > 0.15].content.sam
for text in sample:
  print(text[:200])
```

```
rom: gene@theporch.raider.net (Gene Wright)
Subject: NASA Special Publications for Voyager Mission?
Organization: The MacInteresteds of Nashville, Tn.
Lines: 12

I have two books, both NASA Special P
From: 18084TM@msu.edu (Tom)
Subject: Billsats
X-Added: Forwarded by Space Digest
Organization: [via International Space University]
Original-Sender: isu@VACATION.VENARI.CS.CMU.EDU
Distribution: sci
Li
From: pww@spacsun.rice.edu (Peter Walker)
Subject: Re: The Universe and Black Holes, was Re: 2000 years.....
Organization: I didn't do it, nobody saw me, you can't prove a thing.
Lines: 28

In article
From: da709@cleveland.Freenet.Edu (Stephen Amadei)
Subject: Project Help
Organization: Case Western Reserve University, Cleveland, Ohio (USA)
Lines: 17
```

```
NNTP-Posting-Host: hela.ins.cwru.edu


Hello,

From: dbm0000@tm0006.lerc.nasa.gov (David B. Mckissock)
Subject: Washington Post Article on SSF Redesign
News-Software: VAX/VMS VNEWS 1.41
Nntp-Posting-Host: tm0006.lerc.nasa.gov
Organization: NAS
From: u920496@daimi.aau.dk (Hans Erik Martino Hansen)
Subject: Commercials on the Moon
Organization: DAIMI: Computer Science Department, Aarhus University, Denmark
Lines: 16

I have often thought abou
From: wb8foz@skybridge.SCL.CWRU.Edu (David Lesher)
Subject: Re: No. Re: Space Marketing would be wonderfull.
Organization: NRK Clinic for habitual NetNews abusers - Beltway Annex
Lines: 11
Reply-To: w
From: 18084TM@msu.edu (Tom)
Subject: Solid state vs. tube/analog
X-Added: Forwarded by Space Digest
Organization: [via International Space University]
Original-Sender: isu@VACATION.VENARI.CS.CMU.EDU
D
From: pgf@srl03.cacs.usl.edu (Phil G. Fraering)
Subject: Re: Gamma Ray Bursters. positional stuff.
Organization: Univ. of Southwestern Louisiana
Lines: 24

belgarath@vax1.mankato.msus.edu writes:

>
From: rnichols@cbnewsg.cb.att.com (robert.k.nichols)
Subject: Re: Permanaent Swap File with DOS 6.0 dbldisk
Summary: PageOverCommit=factor
Organization: AT&T
Lines: 50

In article <93059@hydra.gatech.
```

These do seem to be space-related, so let's keep 0.15 as our threshold.

Classification Pipeline

Now that we have a rule for how to see what texts are space related we should incorporate this knowledge into a machine learning pipeline that we can easily use in our future work or production.

For this we are going to use the amazing human-learn library, where you can create rule-based components or even draw things (it really is awesome).

For this we have to freeze the topic model, so that it doesn't get trained when fit() is called on the pipeline

```
pip install human-learn
```

```
from hulearn.classification import FunctionClassifier
from sklearn.pipeline import make_pipeline

# Creating rule for classifying something as a space document
def space_rule(df, threshold=0.15):
    return df["23_satellite_space_launch_nasa"] > threshold

# Freezing topic pipeline
topic_pipeline.freeze = True
classifier = FunctionClassifier(space_rule)
cls_pipeline = make_pipeline(topic_pipeline, classifier).fit(X_train)
```

We now have a classifier for space-related texts, isn't that beautiful? Remember we haven't even touched the labels yet, just used topic models and our own human intuitions.

Evaluation

Just so we can check if this is actually an effective method, let's evaluate our classification pipeline on our test data.

```
from sklearn.metrics import classification_report

y_pred = cls_pipeline.predict(X_test)
print(classification_report(y_test, y_pred))
```

```
              precision    recall  f1-score   support

       False       0.98      0.98      0.98      4475
        True       0.65      0.70      0.68       237

    accuracy                           0.97      4712
   macro avg       0.82      0.84      0.83      4712
weighted avg       0.97      0.97      0.97      4712
```

These are insanely good results considering that we had absolutely zero look at the labels, and the dataset is very unbalanced!!

I suspect that we could tweak this still and get better results with cleaner data, a more informed choice of topic model and potentially more topics so that we can capture more variance in the data.

Nonetheless I considered this method worth sharing and I hope you can use it in your future projects :)))

(((Needless to say if you have labels, please use them, it would be stupid not to)))

Topic Modeling | Unsupervised Learning | Machine Learning | Data Science | Classification

More from the list: "NLP"

Curated by Himanshu Birla

Jon Gi... in Towards Data ...

Characteristics of Word Embeddings

11 min read · Sep 4, 2021

Jon Gi... in Towards Data ...

The Word2vec Hyperparameters

6 min read · Sep 3, 2021

Jon Gi... in

The Word2ve

15 min rea

View list

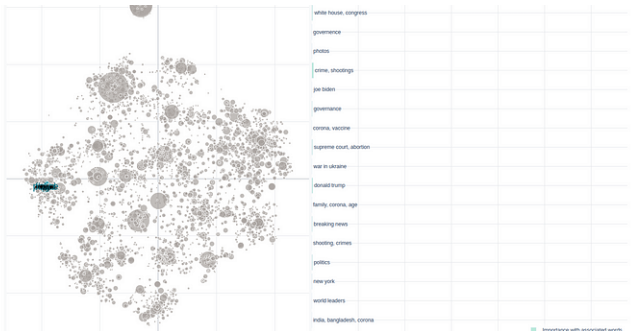
Written by Márton Kardos

Follow

38 Followers

Student of Cognitive Science and Junior Data Scientist at Center for Humanities Computing, Aarhus University.

More from Márton Kardos



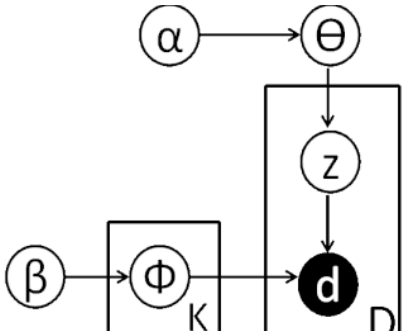
Márton Kardos

Visualizing topic models with topicwizard

Investigate your topic models interactively with beautiful and powerful visualizations.

3 min read · Feb 20

101 6



Márton Kardos

Topic modelling over short texts with tweetopic

A fast, scalable and high-quality approach to modelling short texts and Tweets.

4 min read · Feb 23

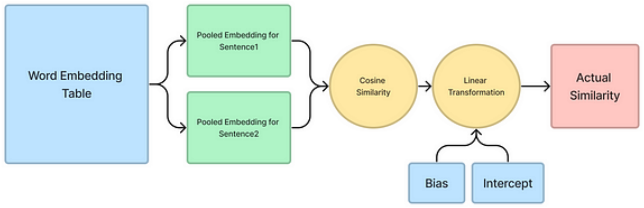
31

Márton Kardos

Zero and Few-Shot Classification with Transformers, LLMs and...

Integrate zero and few-shot classification seamlessly into your scikit-learn workflows.

6 min read · Aug 19



Márton Kardos

Finetuning Static Word Embedding Models

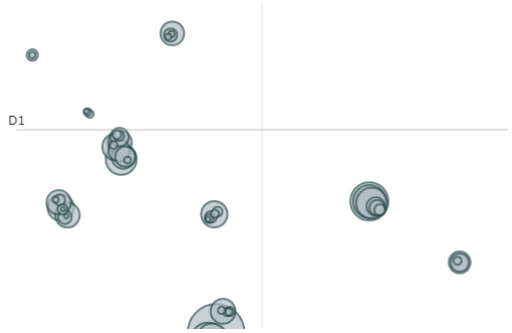
Improve accuracy on downstream tasks for absolutely no hit in performance with a simpl...

7 min read · Sep 12

2 5

See all from Márton Kardos

Recommended from Medium



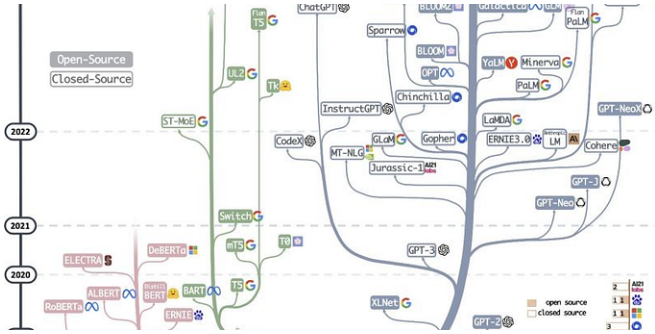
Jawwad Shadman Siddique

Topic Modeling Using BERTopic on Newsgroup Dataset: Python...

We go step by step from creating a google collab workspace to visualizing the cluster o...

7 min read · Jul 5

8



Haifeng Li

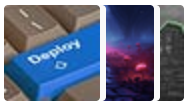
A Tutorial on LLM

Generative artificial intelligence (GenAI), especially ChatGPT, captures everyone's...

15 min read · Sep 14

372

Lists



Predictive Modeling w/ Python

20 stories · 452 saves

Practical Guides to Machine Learning

10 stories · 519 saves

Natural Language Processing

669 stories · 283 saves

New_Reading_List

174 stories · 133 saves

Seffa B

Named Entity Recognition with Transformers: Extracting Metadata

3 min read · Jun 12

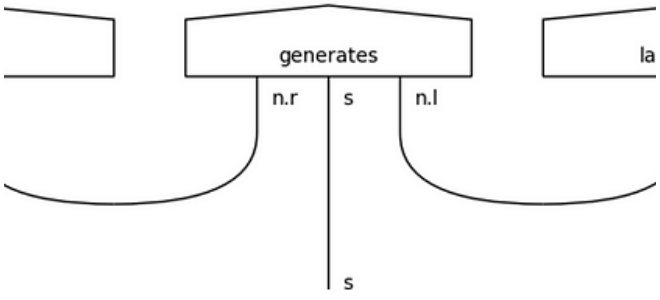
7

Ahmet Taşdemir

Fine-Tuning DistilBERT for Emotion Classification

In this post, we will walk through the process of fine-tuning the DistilBERT model for...

8 min read · Jun 14



Qiskit in Qiskit

An introduction to Quantum Natural Language Processing

Alex Reed

Writing Your First NLP Python Script: From Text Preprocessing t...

By Amin Karamlou, Marcel Pfaffhauser, and James Wootton

10 min read · Nov 4, 2022

227

Welcome to the world of Natural Language Processing (NLP)! NLP is a fascinating field a...

5 min read · Sep 22

56

See more recommendations