DIS

Machine learning *in the wild*

Tales from machine learning after college

DIS 06/02/2023

Duarte O.Carmo

duarteocarmo.com - @duarteocarmo

DIS

Who even are you?

- */du-art/*
- ML/Software Engineer - Contractor
- From Lisbon, based in Copenhagen (Thanks Anders!)
- *Past:* Strategy, Product Management, New Ventures, Management Consulting
- I write code and solve problems end-to-end
- I like running a lot

Today, we'll talk about machine learning from what I've seen out there

- How (I think) ML engineers should work
- 3 example problems from the wild
- "MLOps"
- Learning

- *Opinions*
- *Experiences*

MAGAZINE SPRING 2021 ISSUE / RESEARCH FEATURE

Why So Many Data Science Projects Fail to Deliver

Organizations can gain more business value from advanced analytics by recognizing and overcoming five common obstacles.

Mayur P. Joshi, Ning Su, Robert D. Austin, and Anand K. Sundaram • March 02, 2021

Reading Time: 14 min

1 | How I work

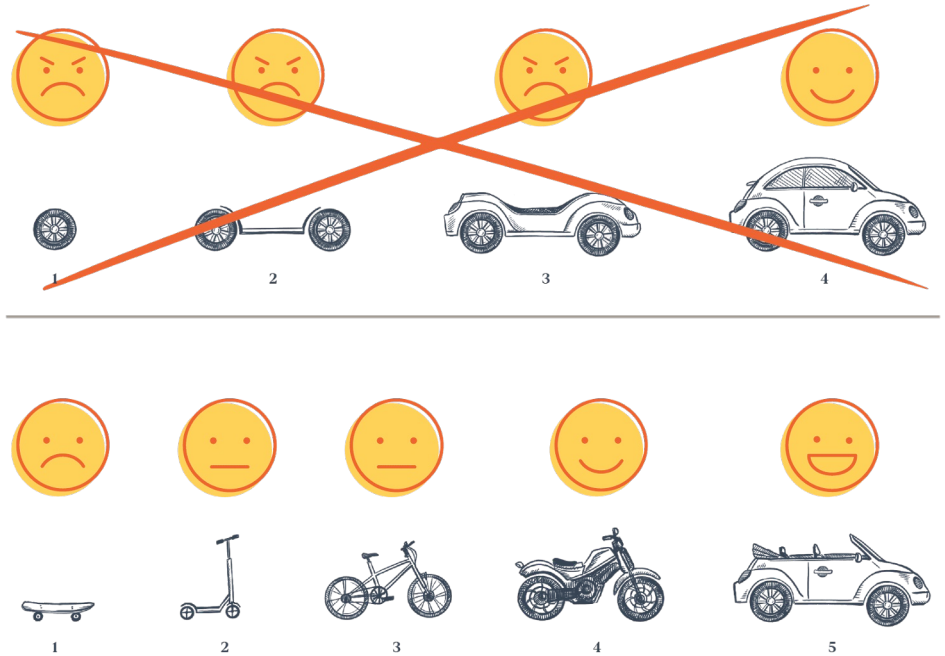
“We need a model”

(you probably don’t)

Don’t start with models, start with people

- Define the business goal, and the success metric
- This is real world (bad) data - not Kaggle: cr*p in, cr*p out
- Start with heuristics, and increase complexity as needed
- Put it out there as fast as possible, then iterate

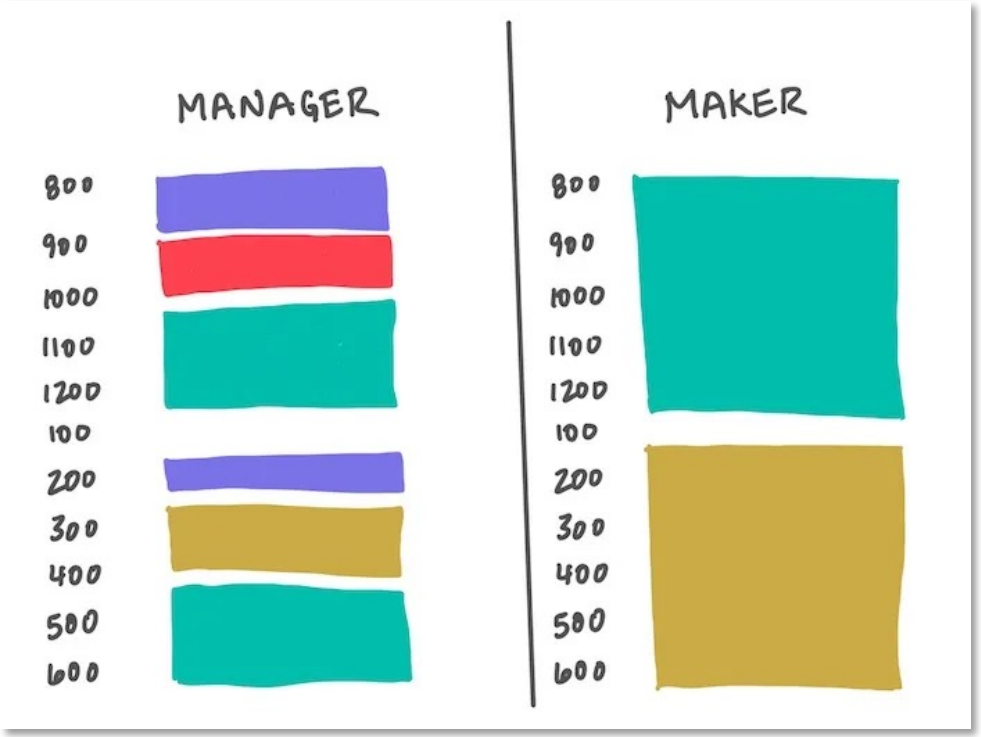
Your goal is to apply research that directly improves users' experiences



- Incredible models are **useless** if not shared with users
- Best model != best solution for the users/business (business metric)
- Quick iterations guarantee you are solving the right problem
- We don't spend too much time in the basement (next slide)

Don’t build in the basement

You are makers at heart – and should treat your schedules like it



- Minimize time in meetings and double down on communication
- Fridays = no meetings
- We are on an emerging tech field, studying is important
- We are builders of things, disruptions are not welcome

2 | Problems

2.1 | Job title classification

Job titles help you find the right people, but we had 38 million

Database with 38 million titles (e.g., “accountant”, “developer ninja”)

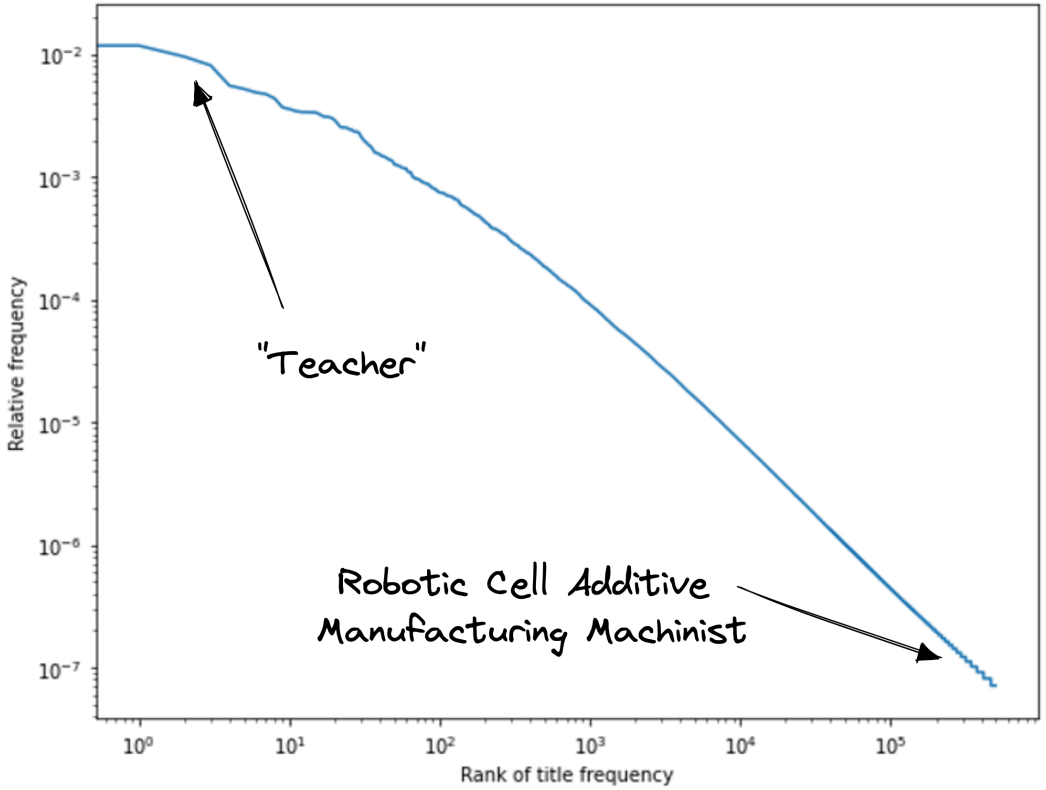
Many ways to search for relevant people

Titles are not easy (e.g., “Product Manager”, “Customer Success Manager”, “Manager”)

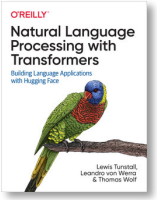
Goal: Categorize job titles into buckets

Most job titles appear millions of times in the DB, we should spend time labelling them

- Not all titles are made equal
- Labelling top 200?
- What can we do with not a lot of data?



DIS



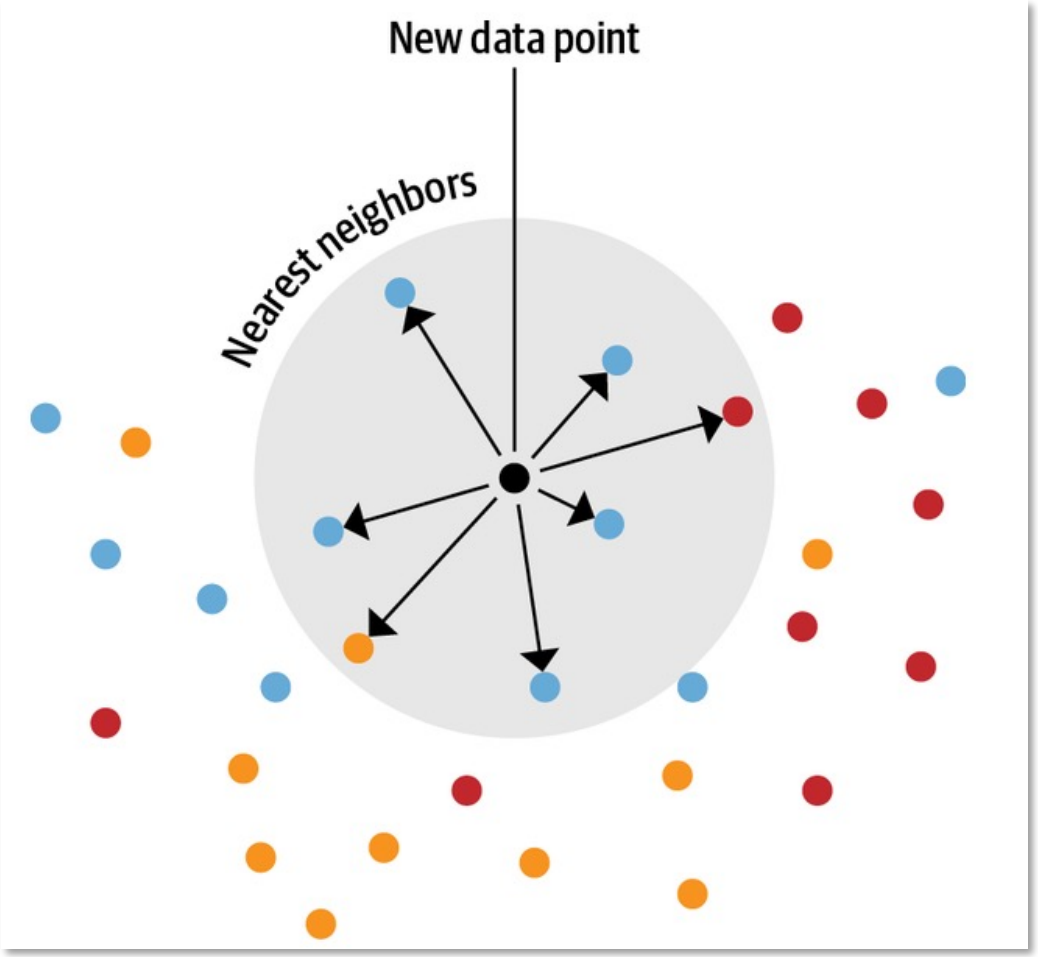


Figure 1: Making a lot with a Little
Credits: Lewis Tunstall, NLP with Transformers O'Reilly

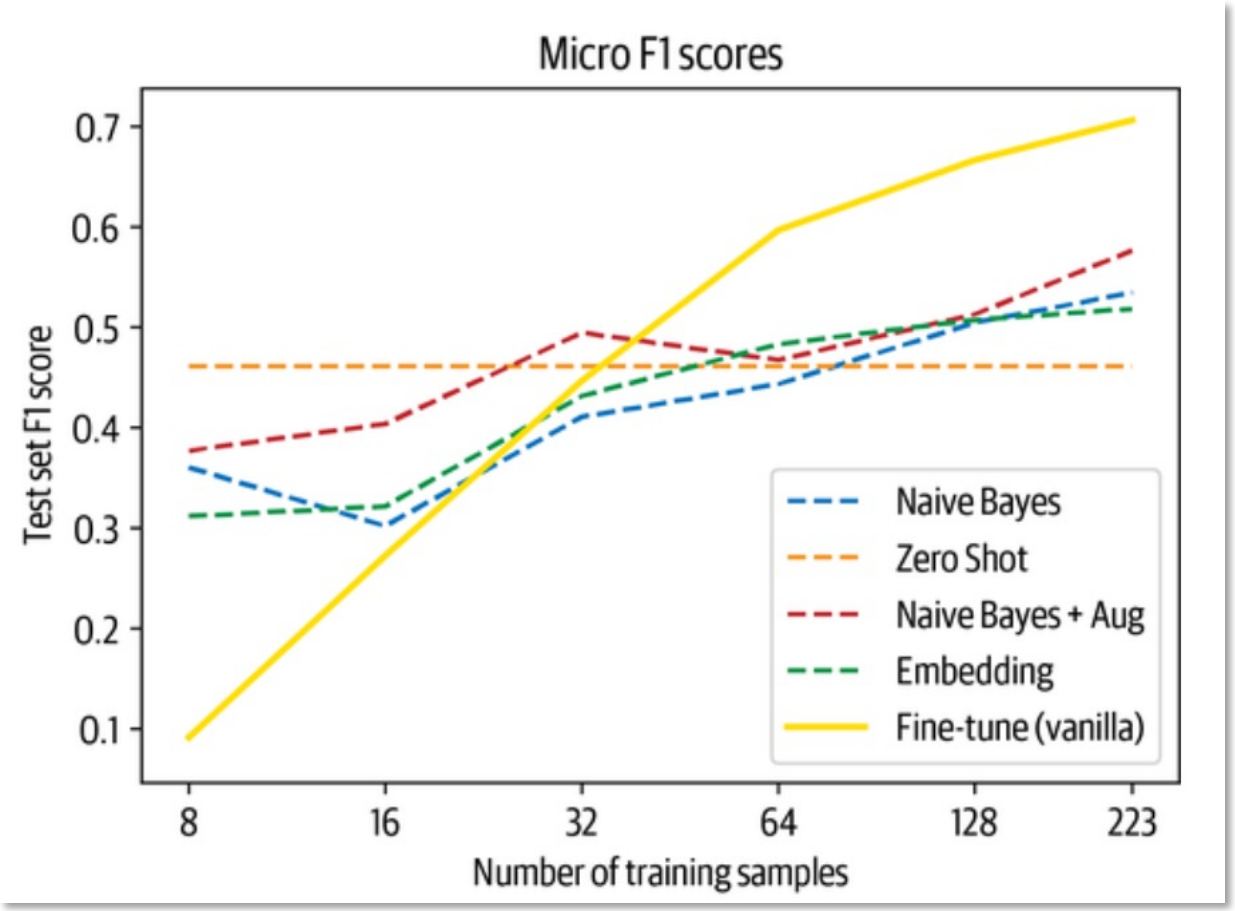
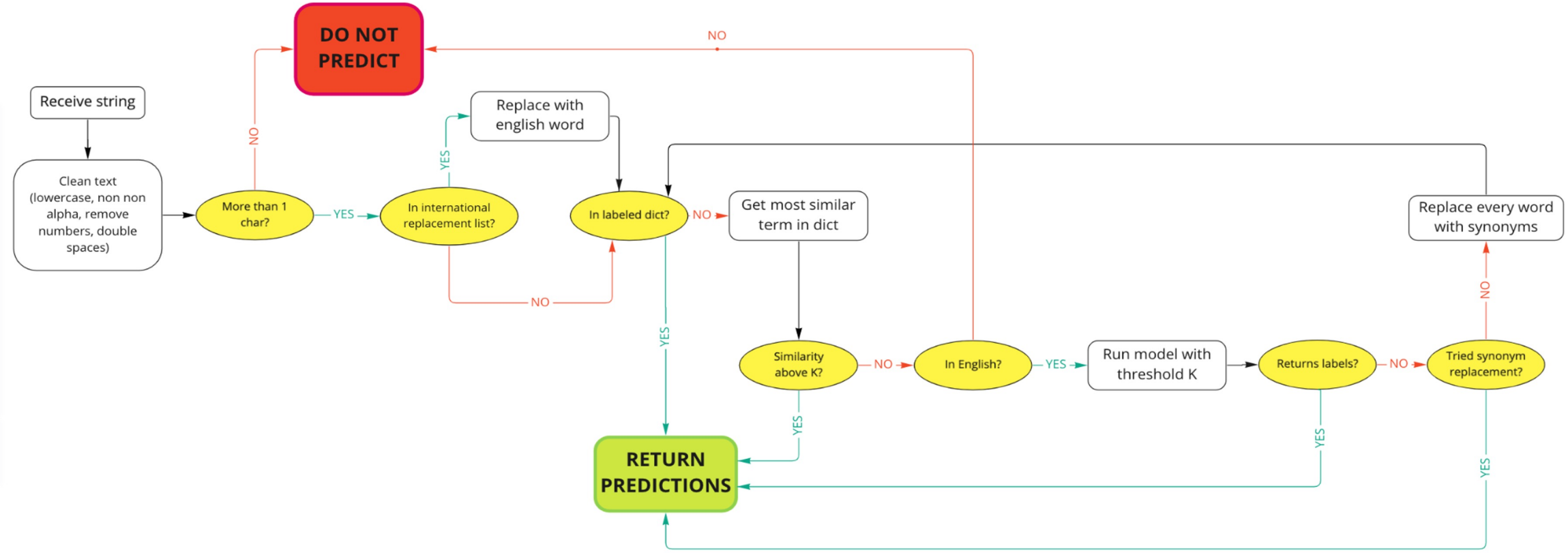


Figure 2: Nearest neighbour lookup
Credits: Lewis Tunstall, NLP with Transformers O'Reilly

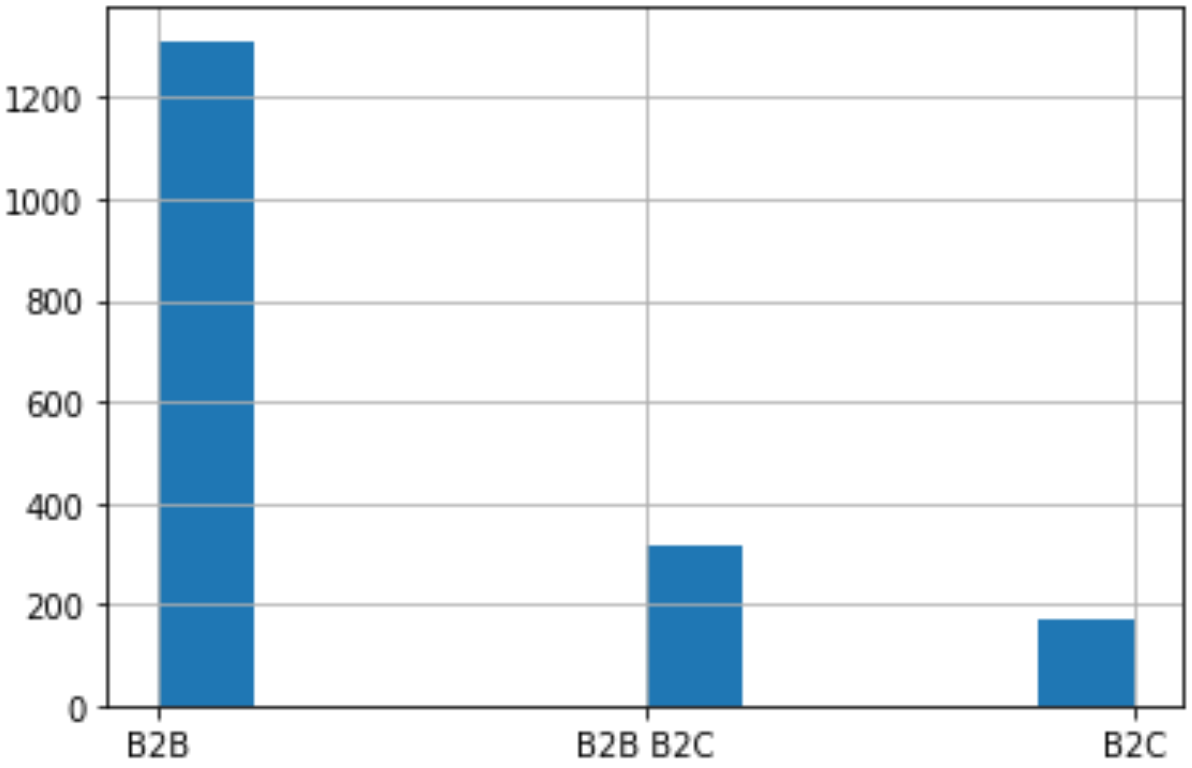
The model is important, but only part of the machinery



DIS

2.2 | B2B/B2C categorization

```
'name’,
'alexa_rank’,
'city’,
'state’,
'country’,
'hq’,
'website’,
'employees_on_linkedin’,
'followers’,
'founded’,
'industry’,
'linkedin_url’,
'overview’,
'ownership_type’,
'sic_codes’,
'size’,
'specialties’,
'total_funding’,
'technologies’,
'company_hubs’,
'events’,
'categories’,
'type’
```



To build a good classifier, you need to be extra careful when defining the problem

Defining the type of problem (e.g., regression, classification, multi-class?)

So many wrong metrics to chose from

Edge cases? (e.g., firefighters, police departments, UNICEF)

How is it going to be used? (what is the cost of wrong?)

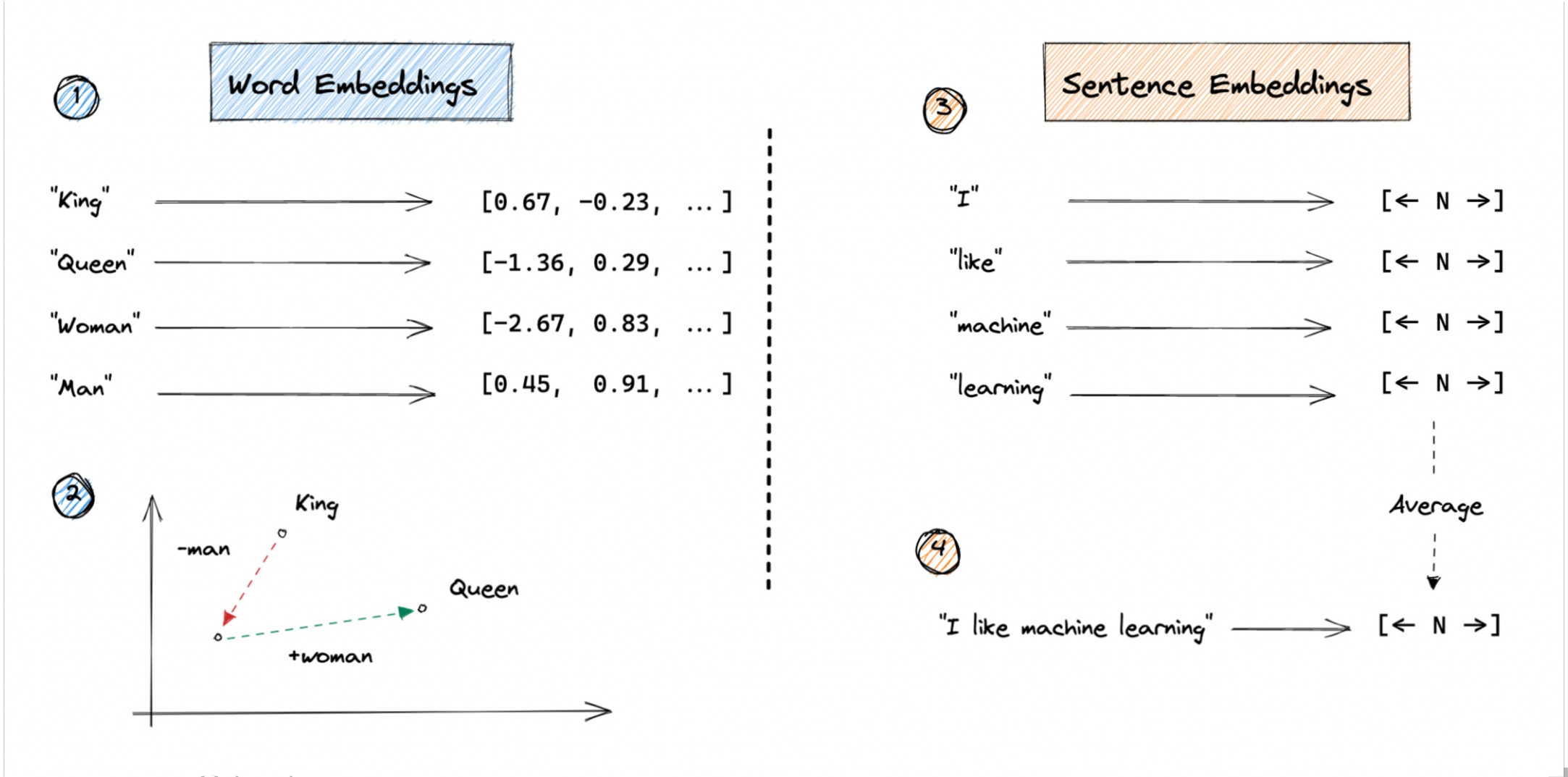
2.3 | Company recommendations

Helping sales teams find their ideal customers

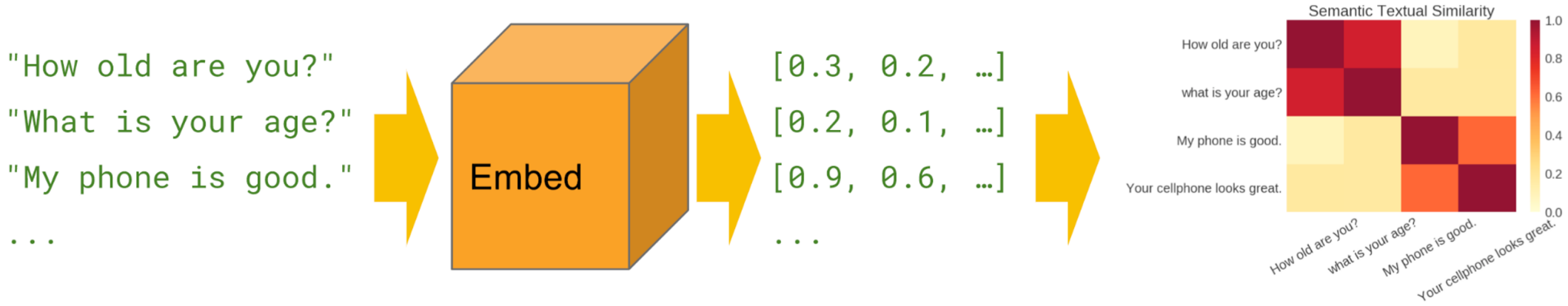
- Lead qualification is manual
- Lots of time spent qualifying
- How can we support this process?

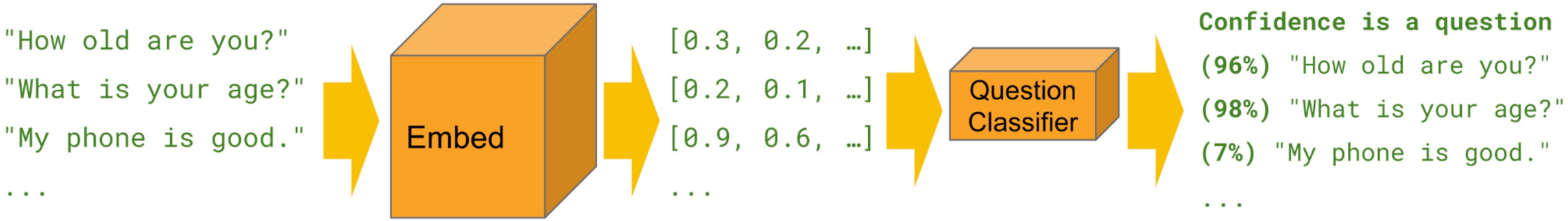
Company Name	Description	Potential Customer?
Novo Nordisk	The Novo nordisk foun..	✅
Facebook	A social media..	❌
Budweiser	We are a bever..	✅
Nike	World leader in..	✅
Google	At Google, we’re..	❌
...	...	...

First, a quick introduction to embeddings

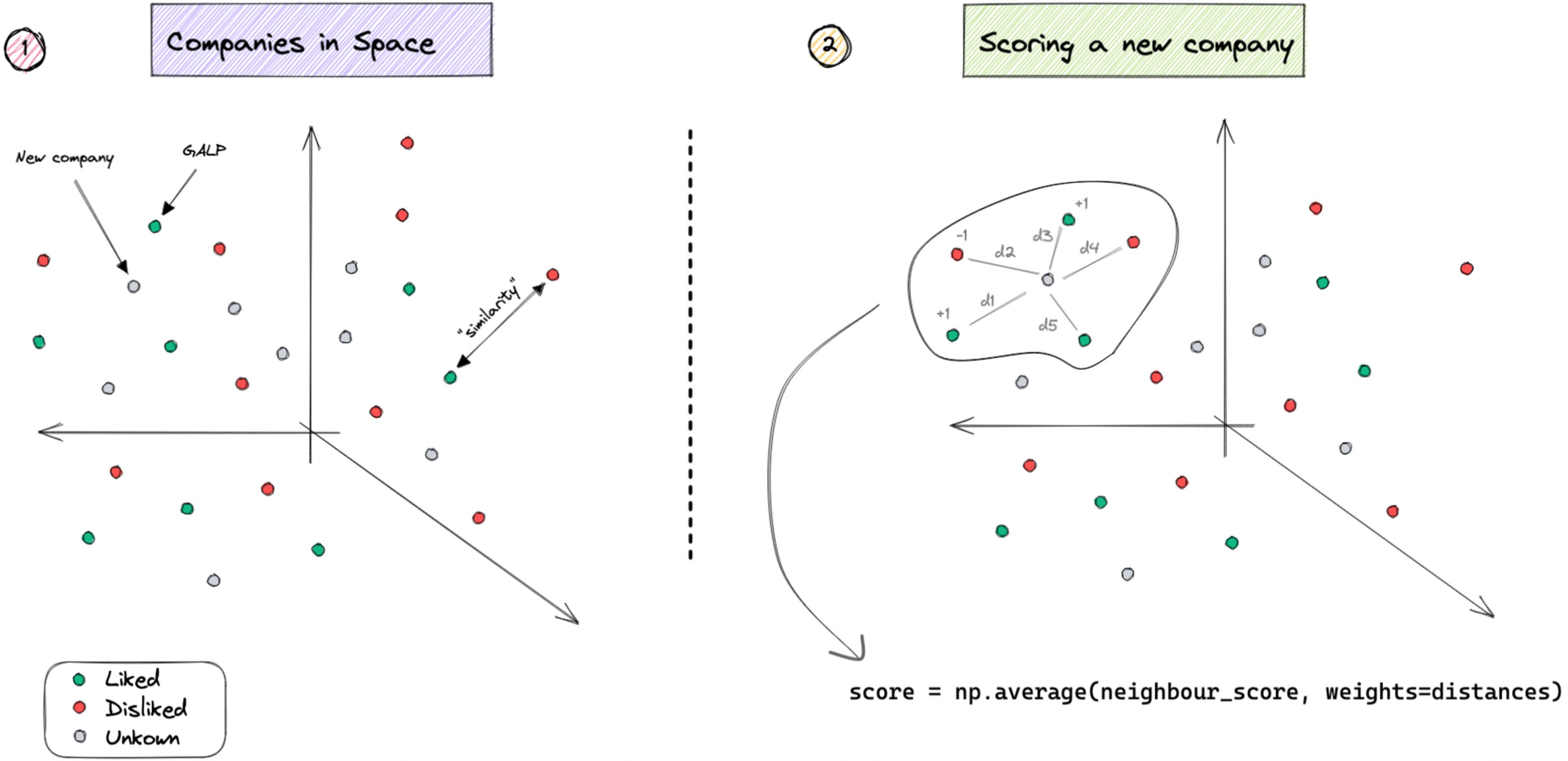


There are a lot of ways to use embeddings in real-world ML problems





DIS




```
score = np.average(neighbour_score, weights=distances)
```

Bonus, find what is wrong on this formula..

DIS

KNN can more *contextual* than a traditional binary classifier

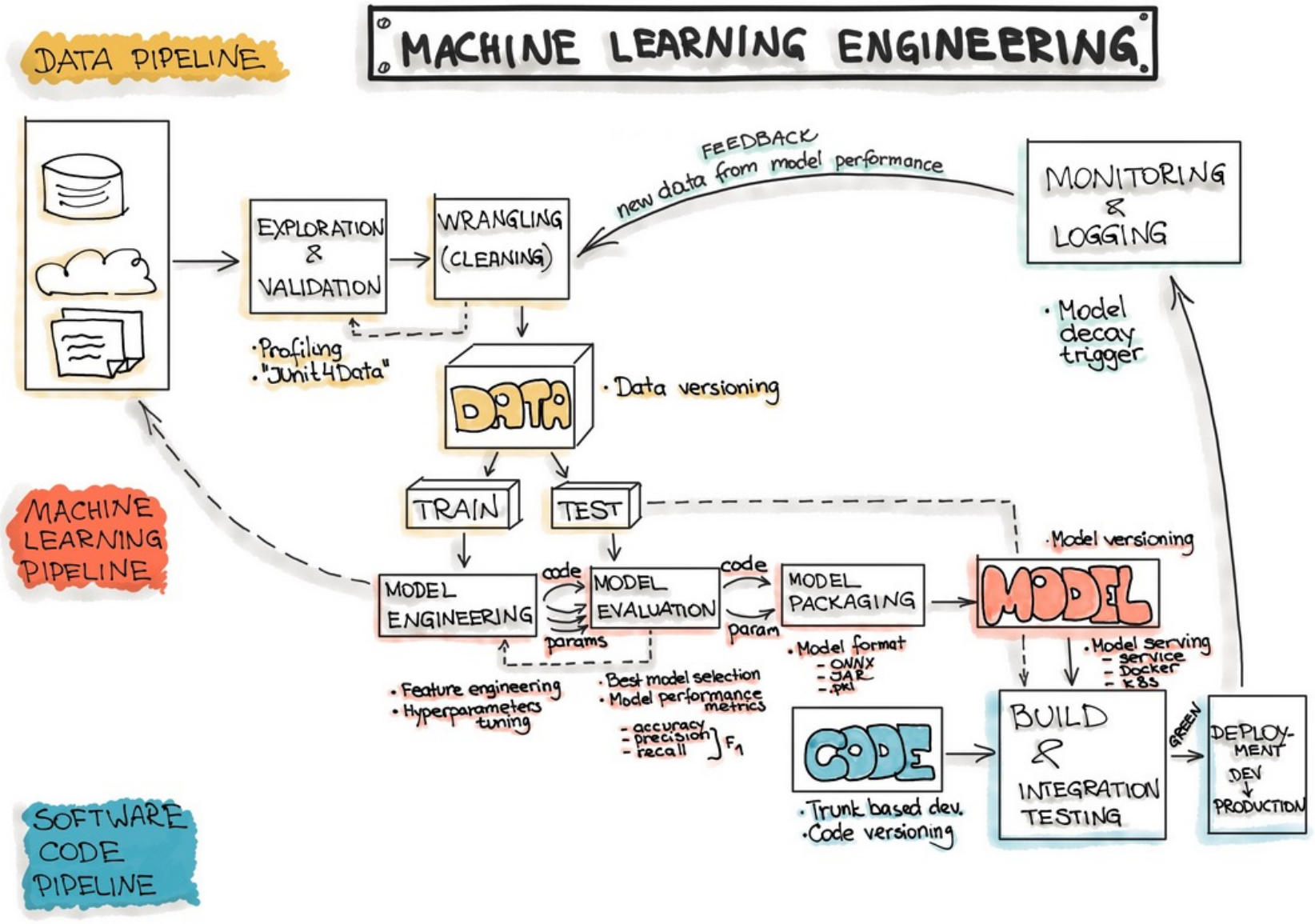
- Embeddings can be re-used - if well chosen
- Recommendations are interpretable
- We can weigh different factors
- Multilingual with unbalanced data
- Generally easier to deploy

Company Name	Most similar	Rec Score
Budweiser	[... N ...]	0.879
Olx	[... N ...]	0.789
Unbabel	[... N ...]	0.678
Novo Nordisk	[... N ...]	0.001
LA Firefighters	[... N ...]	−0.995
...	...	...

Most similar neighbors

Company score

3 | “MLOps”

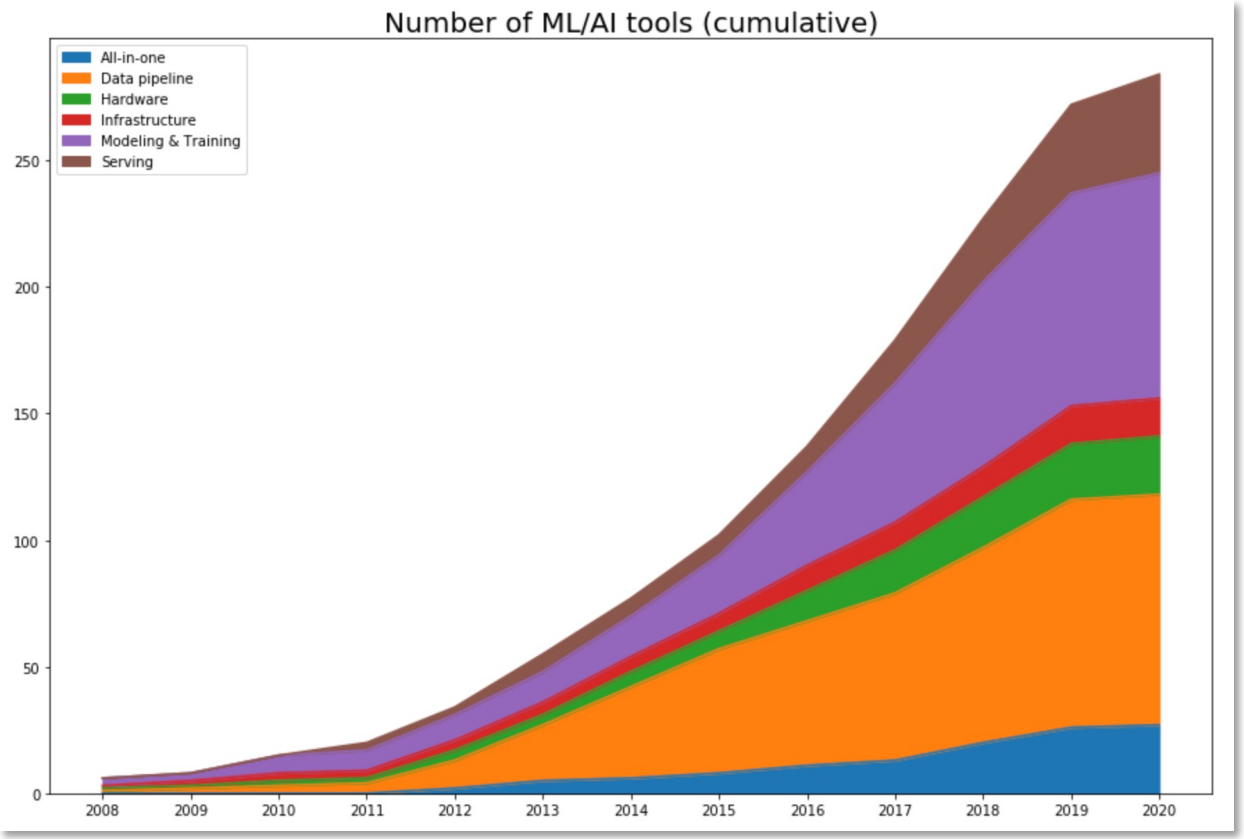


Credits: ml-ops.org

DIS

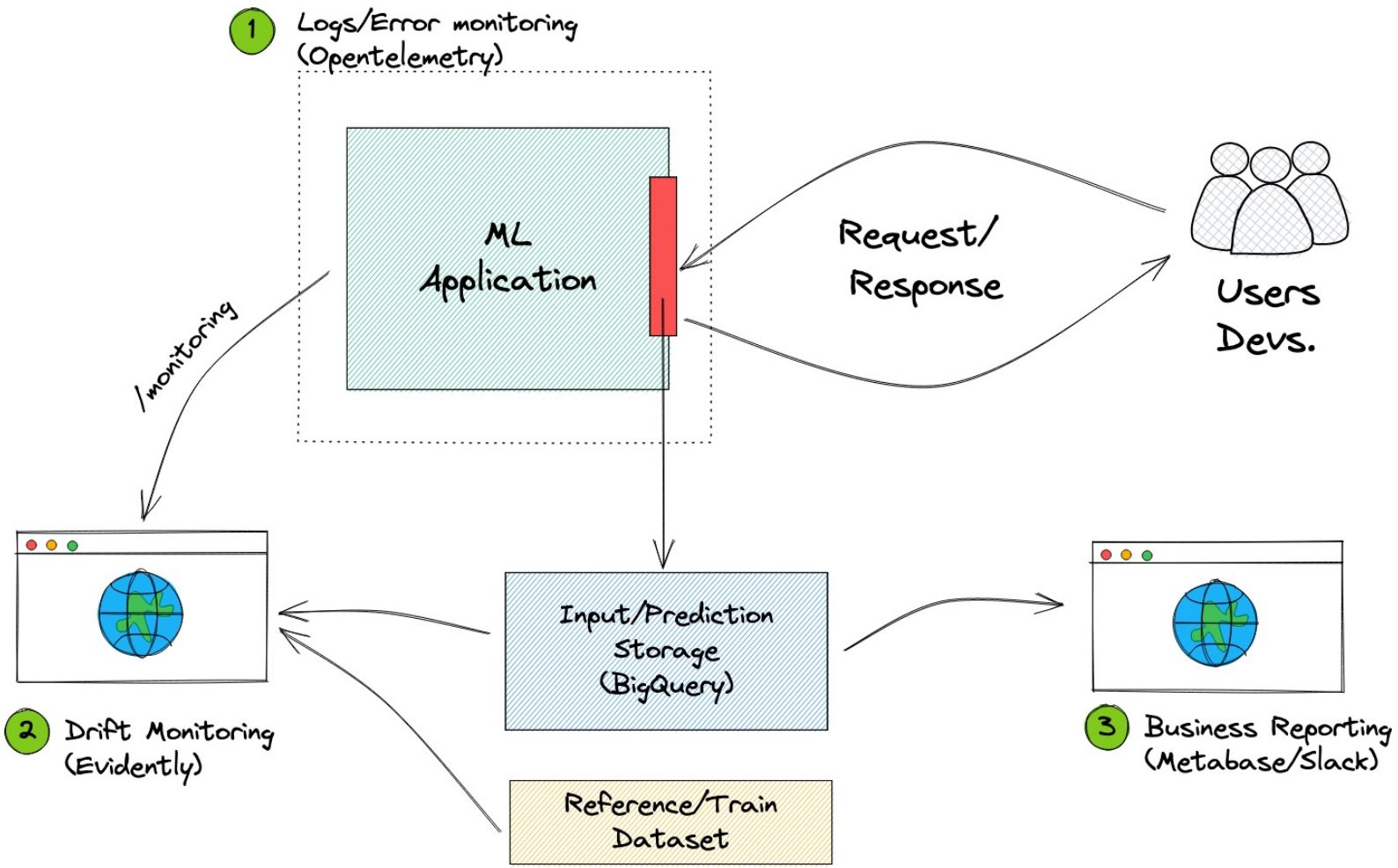
MLOps is not about adopting tools, it's about delivering value

- Gold Rush Age
- FOMO
- Spam emails
- Focus on tools
- 22% have put a model in production
- The real problem: Providing value.



Credits: huyenchip.com

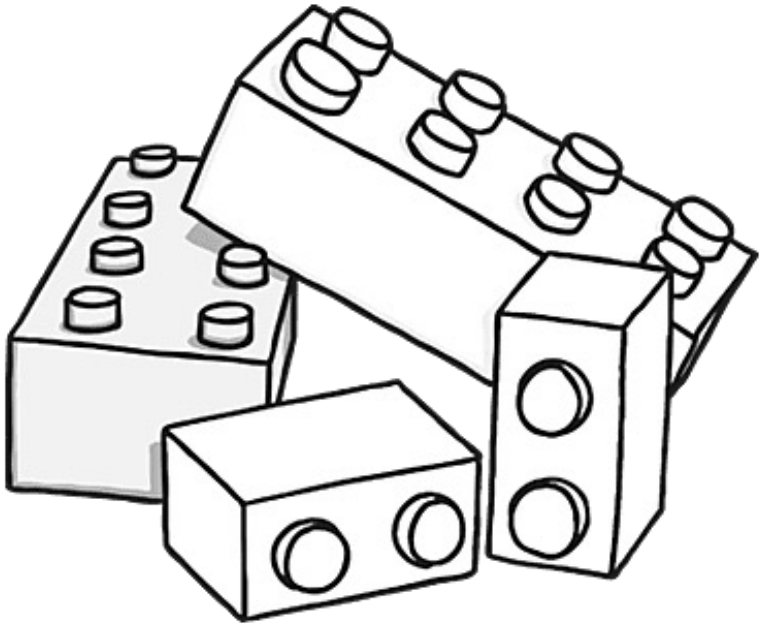
There are essentially 3 different types of monitoring



DIS

4 | Always be learning

First make it work, then make it pretty



- The bare minimum
- Catching all exceptions
- 100% code coverage
- That weird edge case
- Do users care?
- What NOT to write

More ranting: duarteocarmo.com/blog/simple-software

When we start, we have *superpowers*

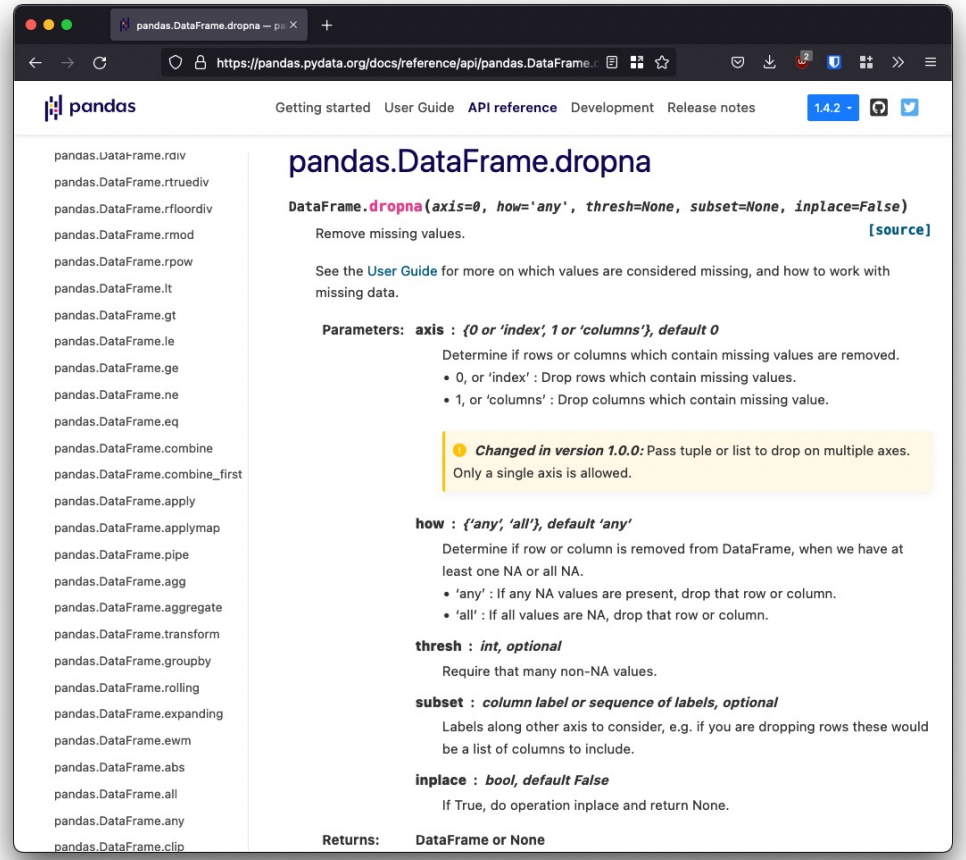
vi similarity.py (vim)

```python
import re
import argparse
import numpy
import pandas
import json
import datetime
import pathlib
from sparse_dot_topn import awesome_cossim_topn
from sklearn.feature_extraction.text import TfidfVectorizer


def read_csv_file(filepath):
    required_columns = ["title", "id"]
    filepath = pathlib.Path(filepath)
    dataframe = pandas.read_csv(filepath)

    if set(list(dataframe)) != set(required_columns):
        raise ValueError(
            f"Make sure that the input csv files have the following columns: {required_columns} "
        )

    names = dataframe["title"]
    ids = dataframe["id"]

    return names, ids


def preprocess(string):
    string = str(string)
    remove_special_chars = re.compile("[^a-zA-Z0-9]+")
    string = string.lower()
    string = string.strip()
    string = remove_special_chars.sub(" ", string).strip()

    return string


def ngrams(string, n=3):
    string = re.sub(r"[,-./]|\s", r"", string)
    ngrams = zip(*[string[i:] for i in range(n)])
    return ["".join(ngram) for ngram in ngrams]


def vectorize(reference, target, analyzer):
    vectorizer = TfidfVectorizer(min_df=1, analyzer=analyzer)
    tfidf_matrix_reference = vectorizer.fit_transform(reference)
    tfidf_matrix_target = vectorizer.transform(target)
```

NORMAL new_version similarity.py python utf-8[unix] 7% 11/157 ln : 1

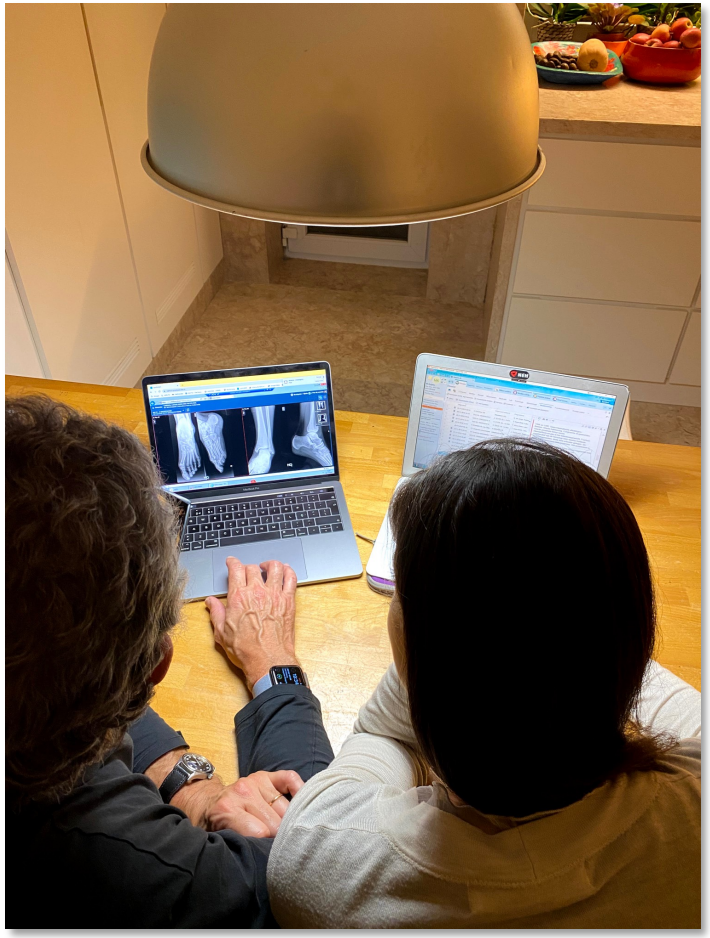
- Autocomplete
- Google
- Stack overflow
- Nails everywhere
- Pip install the world
- But.. We forget quickly

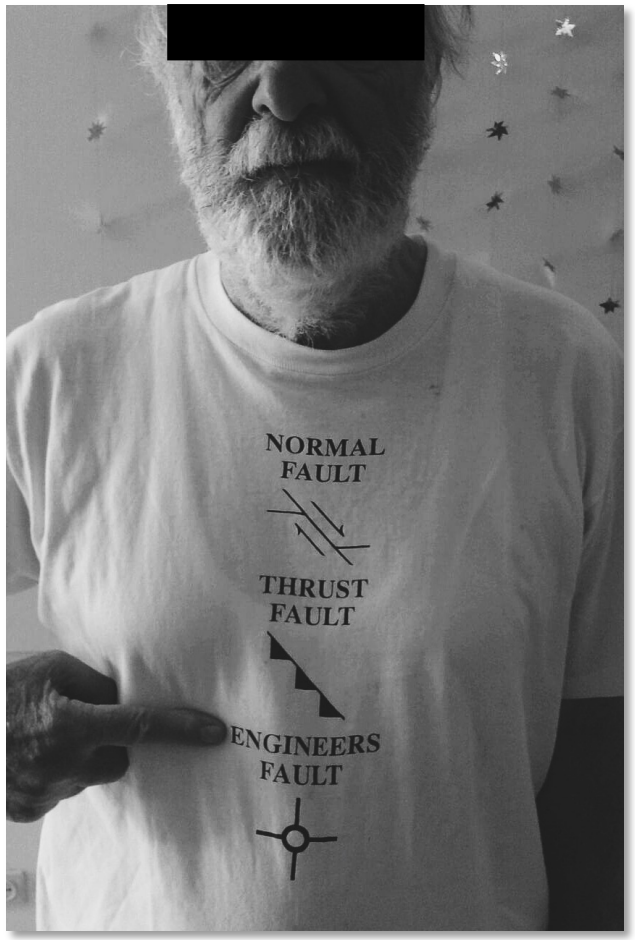
But there’s quite nothing like reading



- What does it do?
- Options?
- Default behaviors
- Maybe I can re-use this
- **It *actually* sticks**

ML is our craft

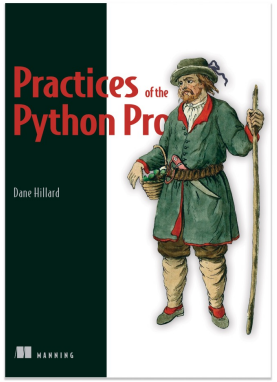


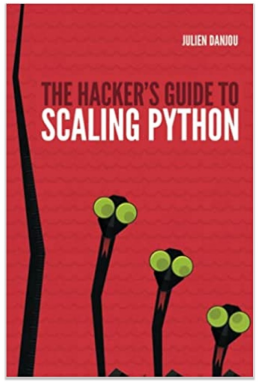

NORMAL
FAULT
THRUST
FAULT
ENGINEERS
FAULT

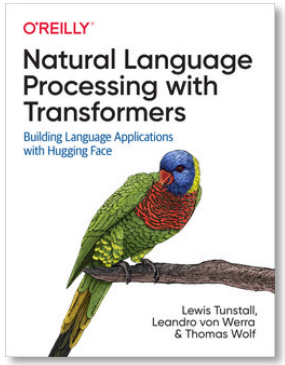

EINSTEIN
TINHA
RAZÃO?
Testando
a Teoria
da Relatividade
Geral
Clifford
M.Will
NOVO
ATLAS
GEOGRÁFICO

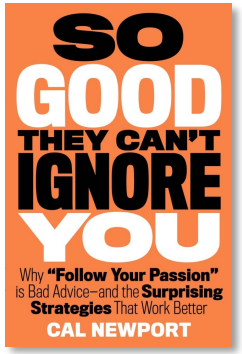
We should be masters of our craft

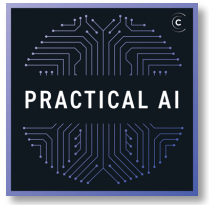
- Study
- Stay up-to-date
- Lean regularly
- Build things
- Give back and write











An OCD list of resources

Books

Practices of the Python Pro

Hacker’s guide to scaling Python

Designing Data-Intensive Applications

Serious Python

Tutorials

Flask Mega-tutorial

RealPython

Stack Abuse

Kaggle + GitHub

YouTube

CodingTech

Sentdex

Abhishek Thakur

MLOPs Community

Podcasts

Talk Python to Me

Python Bytes

Podcast.__init__

Practical AI

News

PyCoder’s Weekly

Medium

Awesome Python Weekly

Reddit RSS

...

Thank you, questions?