mgm

*"Using a simple tool to solve a complex problem does not result in a simple solution."* Larry Wall

# Classifying unstructured text

## Deterministic and machine learning approaches

Stephanie Fischer
Dr. Christian Winkler

**Hamburg** München Berlin Köln Leipzig

Apache Big Data Sevilla , 15th November 2016

# Agenda

01 About us
02 Text statistics
03 Categories
04 Text classification
05 Conclusion and outlook

# Speaker

**Stephanie Fischer**

Product Owner Text Analytics
Big Data, Agile & Change
mgm consulting partners

**Dr. Christian Winkler**

Enterprise Architect
Big Data, Data Science
mgm technology partners

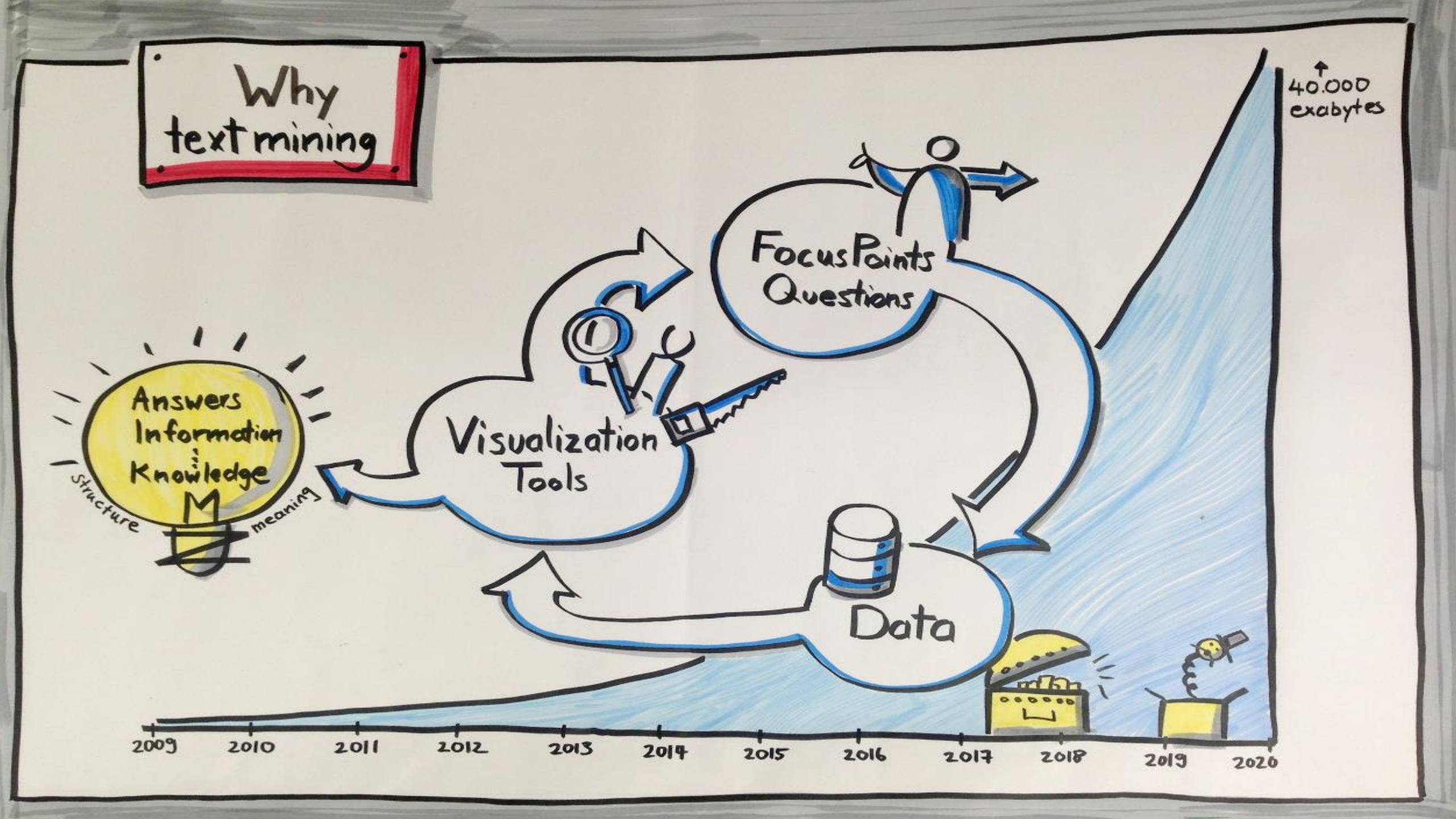
01

# About us

# Stephanie and Christian according to their browser history

02

# Text statistics


Why text mining
Focus Points Questions
Visualization Tools
Data
Answers Information Knowledge
Structure
meaning
40.000 exabytes
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020

# Comparing **word frequency** of news from Reuters, Telegraph, Aljazeera


## Reuters World News

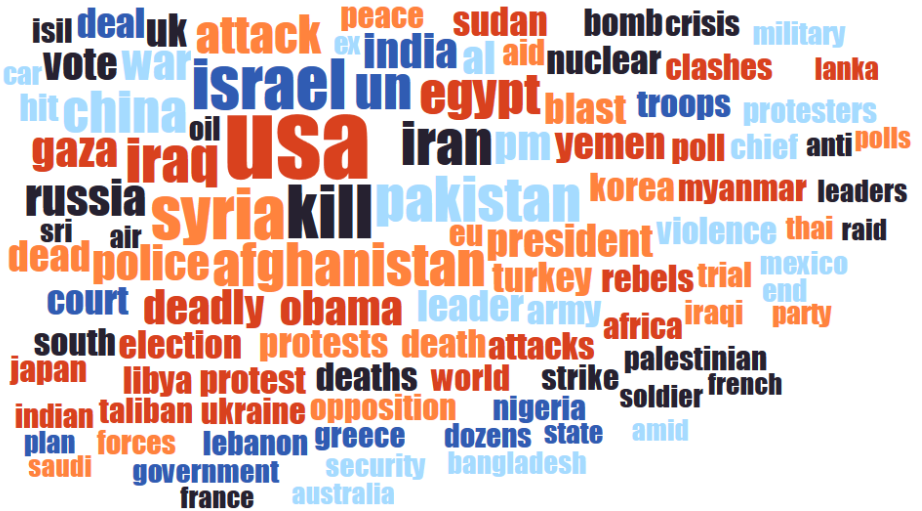
# 163,919 headlines

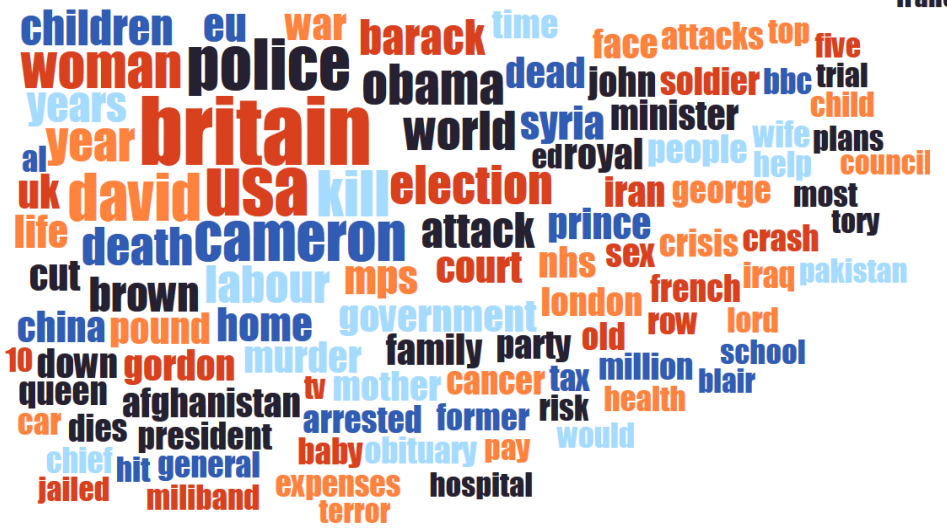
9 years

## Telegraph

# 958,996 headlines

9.5 years




## Aljazeera

# 94,309 headlines

8.5 years

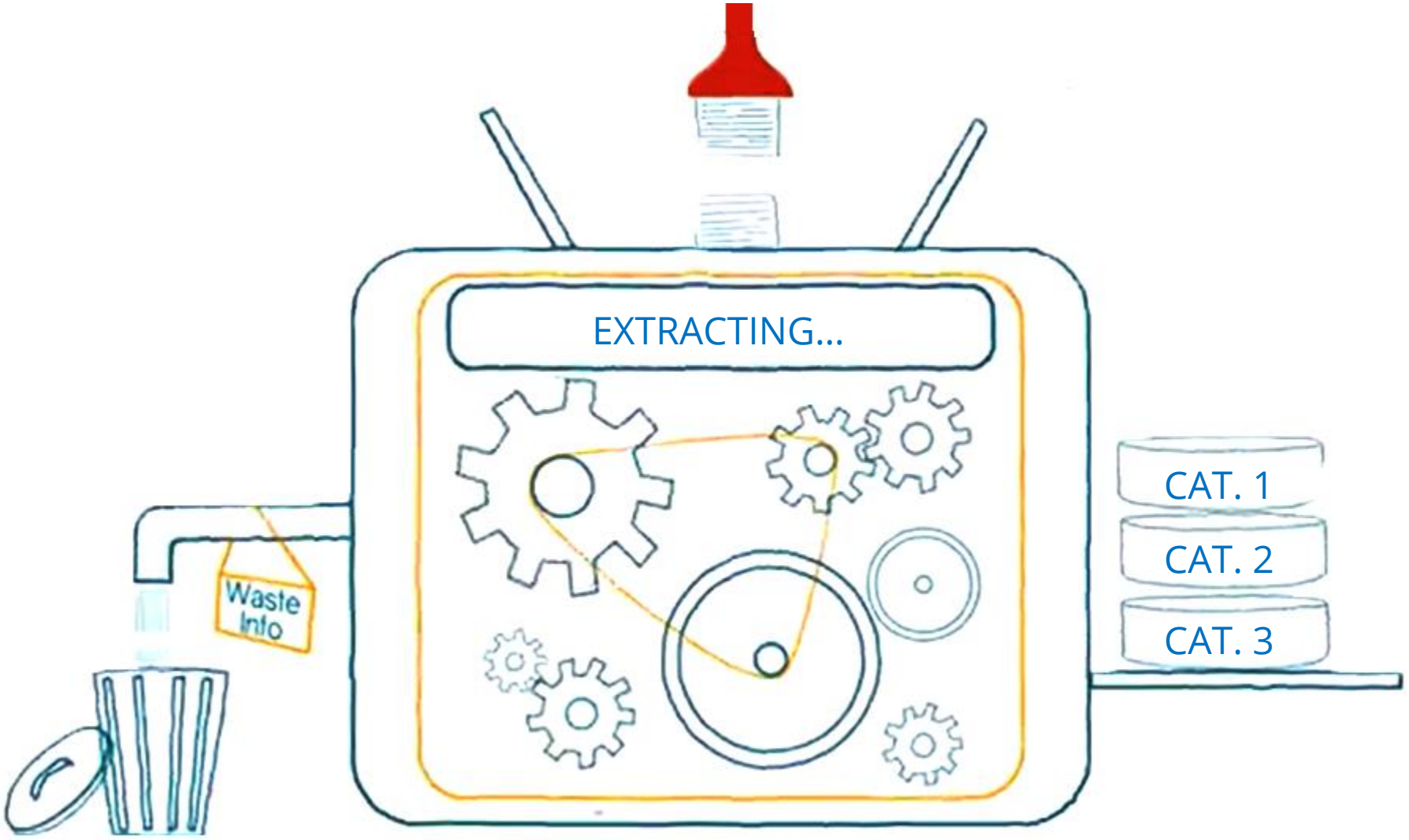
mgm

Visualizations created with Apache Solr and D3.js, see our talk from Apache Big Data Vancouver 2016 here: https://bigdata.mgm-tp.com/apache/

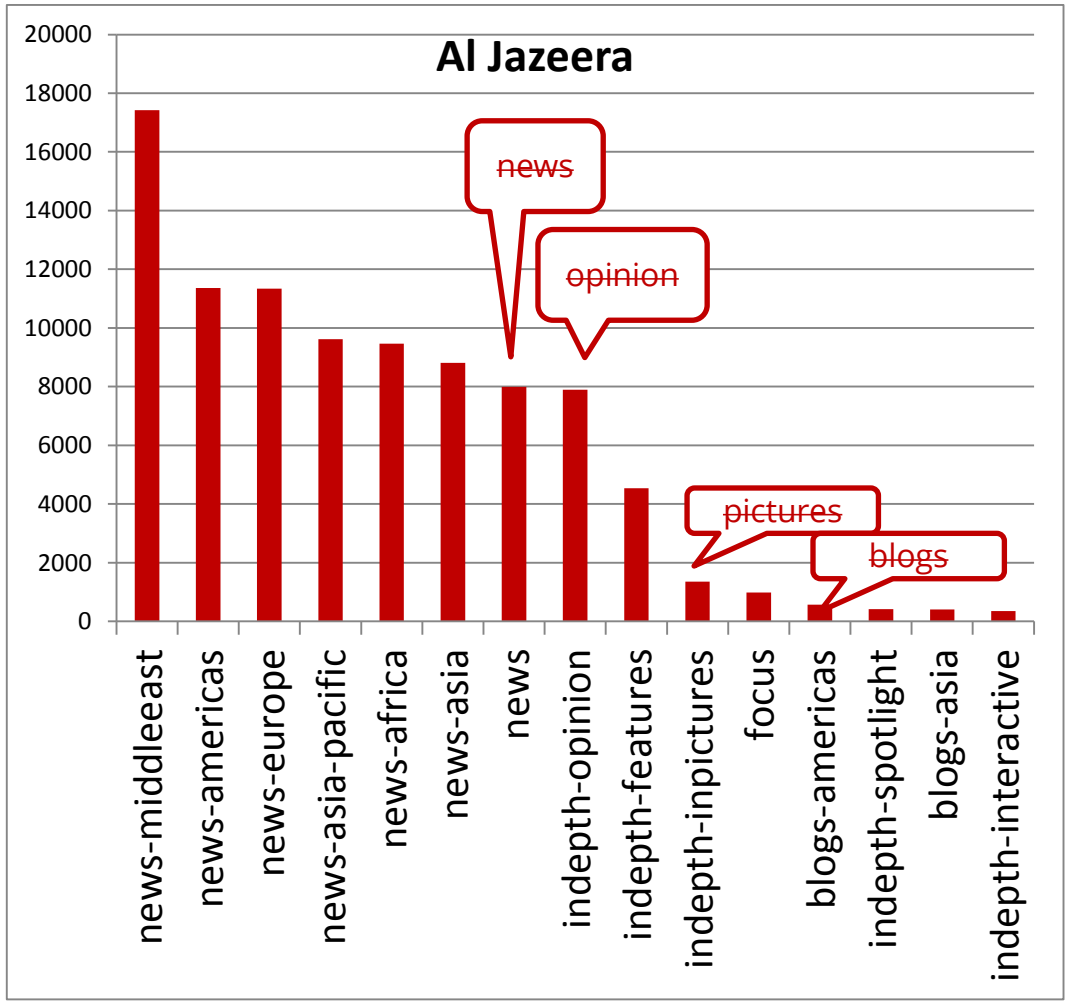
03

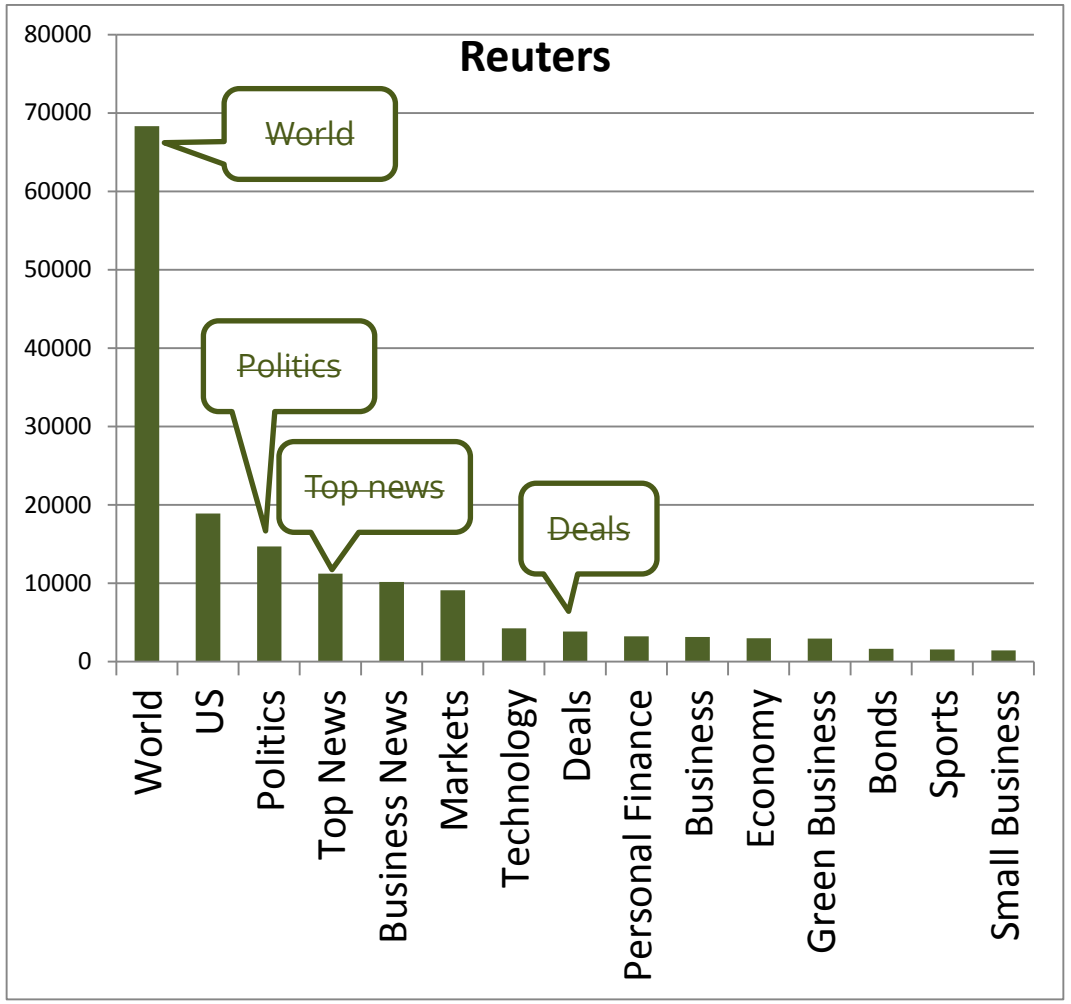
# Categories

# Finding meaningful categories. Each text is different. Challenge accepted!



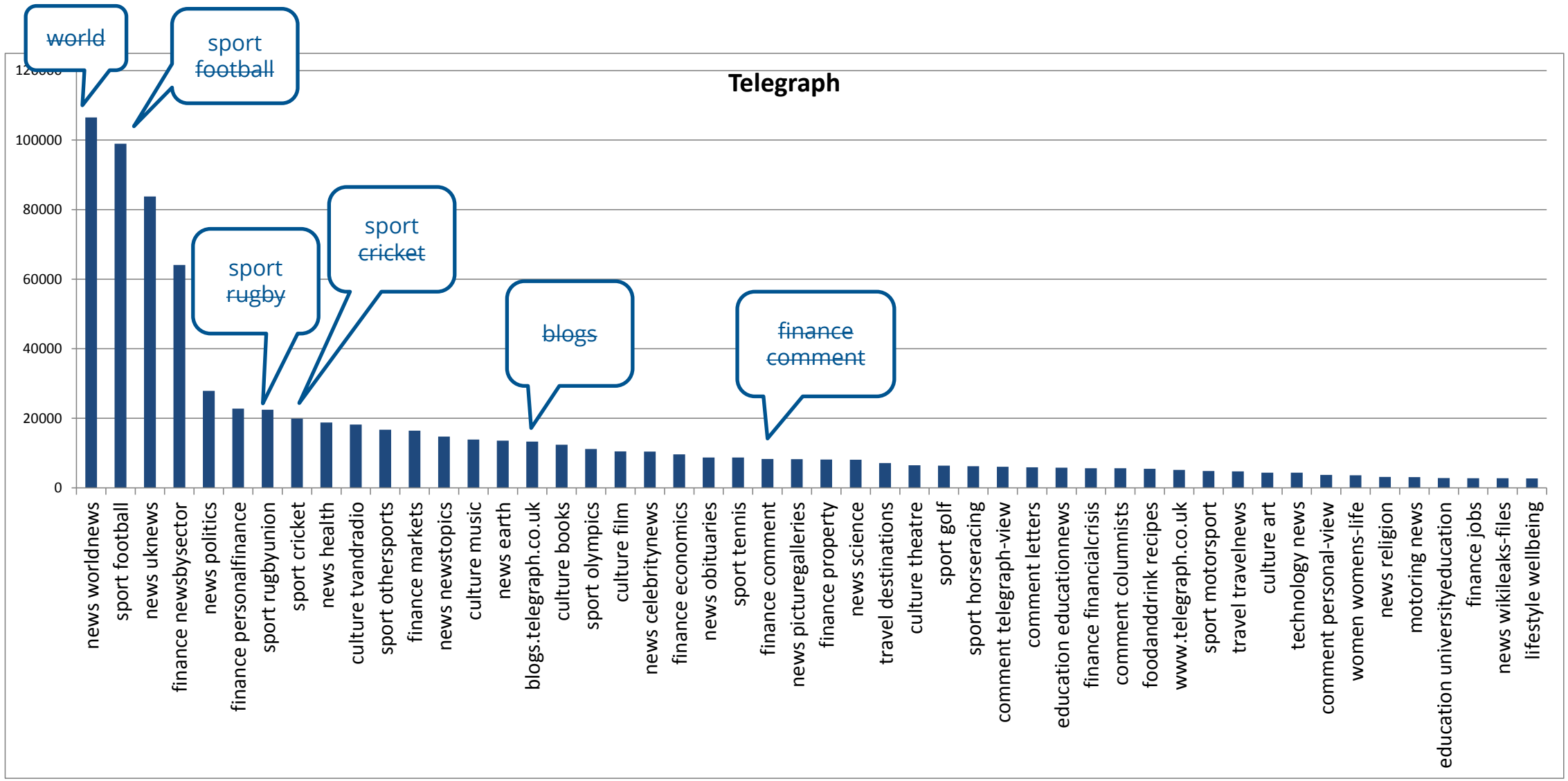
# Comparing pre-defined categories of Al Jazeera, Reuters…




mgm

# … and the Telegraph categories

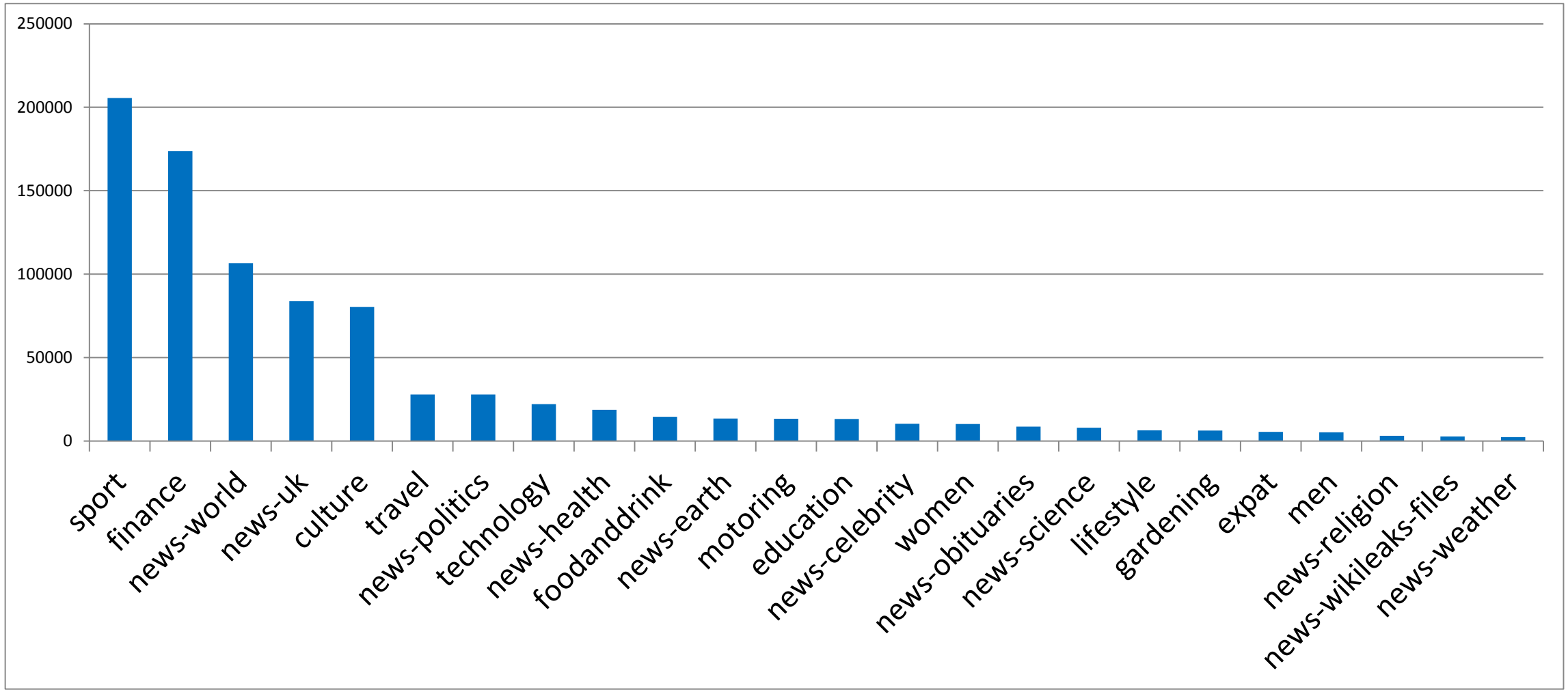



mgm

It's not so easy.

mgm

# Our selection: Functionally relevant, mutually exclusive categories derived from Telegraph categories




mgm

Finding **meaningful categories** for the Telegraph News was fun!

Lets go on and do a whole **text classification experiment.** Our **aim** is to **classify 1 million Telegraph News documents** with an ML algorithm.

While doing this we want to **find out**...

... if a **ML algorithm** will be able to classify the Telegraph news documents

... what are the **steps** we need to work out in order to make the ML algorithm work?

Handy for us: We will be able to **train** the **ML algorithm with the pre-classified data set** of the Telegraph News!

mgm

04

# Text classification

# Typical text classification projects and our experiment set-up

Typical set-up:
no classification scheme, no classified data

- Choose data to be classified
- Manually classify chosen data set
- Train ML algorithm with classified data set
- Apply trained ML algorithm to complete data set
- Manual QA data set samples

Our Telegraph experiment with pre-classified documents

- Choose data to be classified
- Get already existing classifications for chosen data
- Train ML algorithm with classified data set
- Apply trained ML algorithm to complete data set
- Automatic QA complete data set

## Advantages for us:

- ✓ No manual classification & QA necessary
- ✓ Existing classification scheme
- ✓ Playground easily set up
- ✓ Free to choose both manual data set & categories

mgm

# Our experiment for the next 30 minutes

Typical set-up:
no classification scheme, no classified data

- Choose data to be classified
- Manually classify chosen data set
- Train ML algorithm with classified data set
- Apply trained ML algorithm to complete data set
- Manual QA data set samples

Our Telegraph experiment with pre-classified documents

- Choose data to be classified
- Get already existing classifications for chosen data
- Train ML algorithm with classified data set
- Apply trained ML algorithm to complete data set
- Automatic QA complete data set

Our aims in the next 30 minutes:

- ✓ Train & apply the ML algorithm to 1 of Telegraph News
- ✓ See how well ML performs

mgm

This process sounds easy and very structured. The people in the audience who have already done text classification projects probably now that in reality, data can become **pretty challenging**.

The next slides show you the process of **how** we classified 1 Million  Telegraph news.

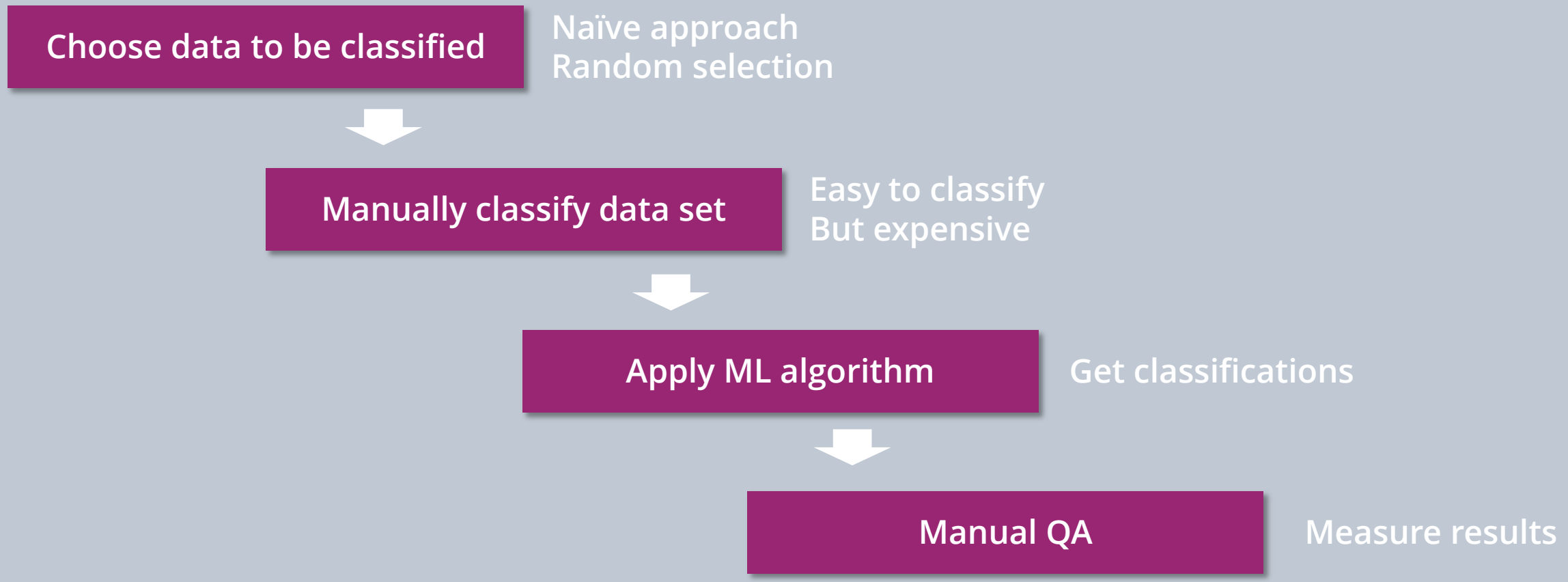
What is the **reality** we deal with?

And what are **good practices/our learnings?**

# The devil is in the data

mgm

# Getting started: Preparing data for and executing ML



The result is **BAD!**

# WHY?

Lets take a step back and find out:

How does **ML WORK?**

How can I **MEASURE** its results?

mgm

# ML algorithm explained – Support Vector Machine (SVM)

Machine learning is linear algebra

- Need to discretize first

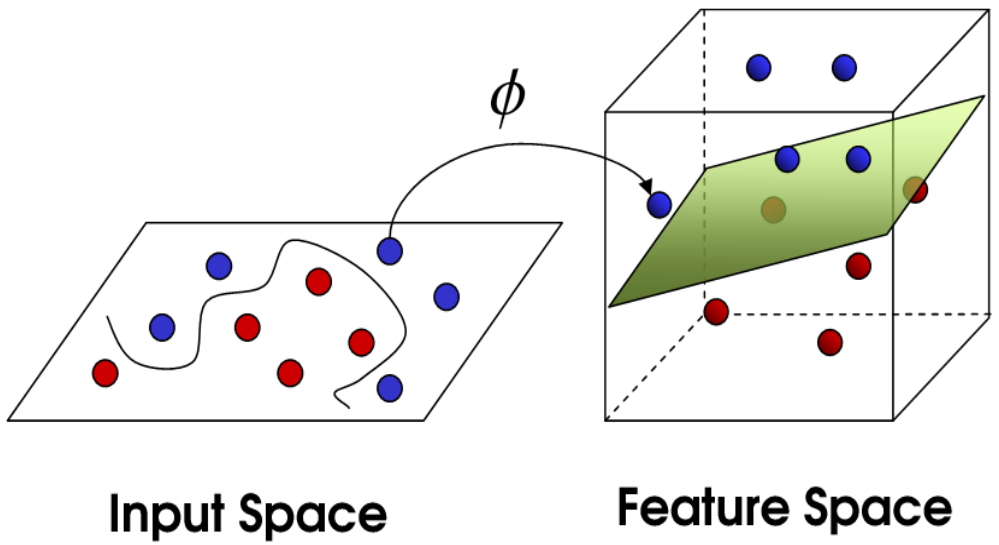
Categories are already discrete

More complicated for text

- Bag of words = detect words
- TF/IDF matrix = use document and total frequency

Many different possible learning models

- Support Vector Machines (most popular)
- Neural Network
- Random forest
- Decision tree



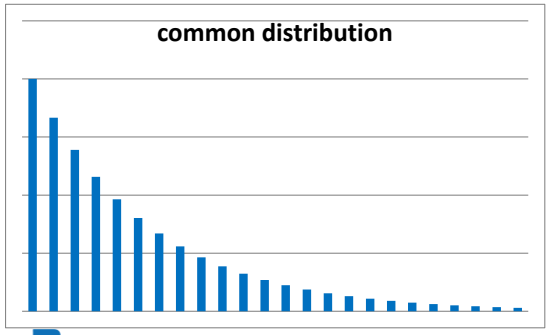

Support Vector Machine

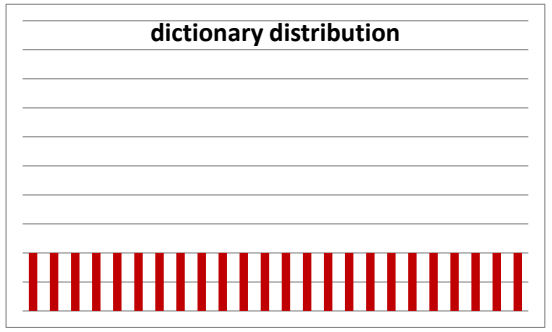
mgm

# Preparation of manually classified set

## Choosing set for manual classification

- Select documents with highest word variability
  - Metric:
    Word heterogenity
    = Number of words in all documents
    (→ stopwords)
  - Even distribution
  - Long tail distribution
    (→ many, many words use infrequently )
- Complicated: knapsack-like problem
- Use an approximate approach (like genetic algorithm)
- Crucial for all following tasks




**1. Good situation:**

The manually classified data set contains all the words of the complete data set.

| Word heterogeneity in manual set | | | Word heterogeneity complete data set | | |
|---|---|---|---|---|---|
| w01 | w02 | w03 | w01 | w02 | w03 |
| w04 | w05 | w06 | w04 | w05 | w06 |
| w07 | w08 | w09 | w07 | w08 | w09 |
| w10 | w11 | w12 | w10 | w11 | w12 |
| w13 | w14 | w15 | w13 | w14 | w15 |
| w16 | w17 | w18 | w16 | w17 | w18 |

**2. Not so good situation:**

The manually classified data contains only a fraction of all the words in the complete data set

| | | |
|---|---|---|
| w19 | w20 | w21 |
| w22 | w23 | w24 |
| w25 | w26 | w27 |
| w28 | w29 | w30 |
| w31 | w32 | w33 |
| ... | ... | w99 |

Complete set

mgm

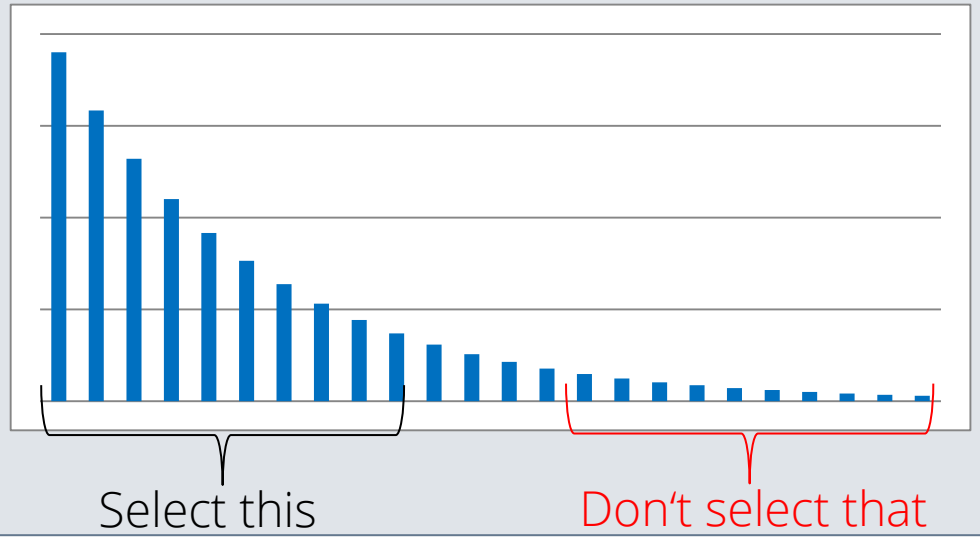
# Intelligently choose data set to be classified manually

## Final data set available

- Choose training data set in a way to create maximal word overlap with complete data set
- $W_M$ = { words in training set }
  $W_C$ = { words in complete set }
  find maximum for $| W_C \cap W_M | = | W_M |$
- Improved approach: choose training set to minimize headlines with unknown words in complete data set
- Find minimum for $|C \cap \overline{W_M}|$
- More complicated, but worth it

## Final data set not available

- Optimize for high variabitlity and high usage




mgm

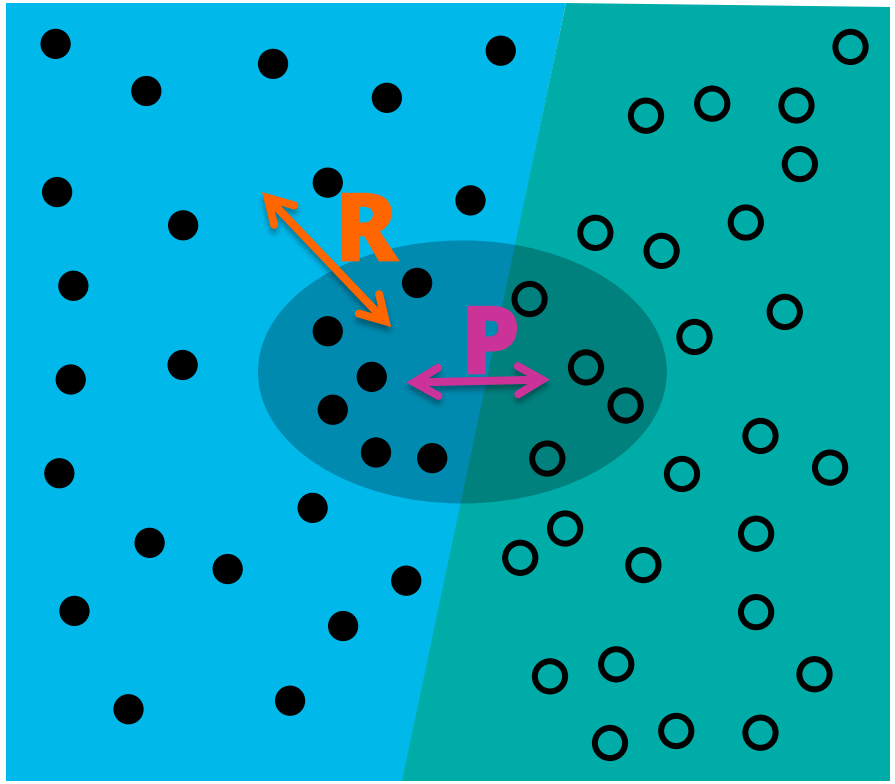
# Measure classification quality: precision and recall

**Precision**

- „positive predictive value“
- Precision is the probability that a (randomly selected) retrieved document is classified correctly

**Recall**

- Sensitivity or „true positive rate“
- Recall is the probability that a (randomly selected) classified document is found




**Example**

- Africa has very high precision for category „Africa“, but bad sensitivity (recall)

# Now we know why the naive approach of preparing data for and executing ML is not enough. Lets try the following instead…

**Choose data to be classified**

Calculate text metrics
Define goals
Use optimization

**Manually classify data set**

Easy to create
But expensive
Optimal set

**Apply ML algorithm**

Get classifications
Training and test data set
Calculate Precision + Recall
Crossfolding, use different algorithms

**Manual QA**

Better results

**Necessary steps to successfully apply ML**

- Measure quality of ML
- Training set and test set
- Precision and recall
- Apply ML to whole data set
- Manual QA

**Attention:**
ML remembers words
→ It can only classify text with known words

# What we have done & achieved so far

1. **Data cleaning and preparation:**
   **Docs with same headline but different classification removed**

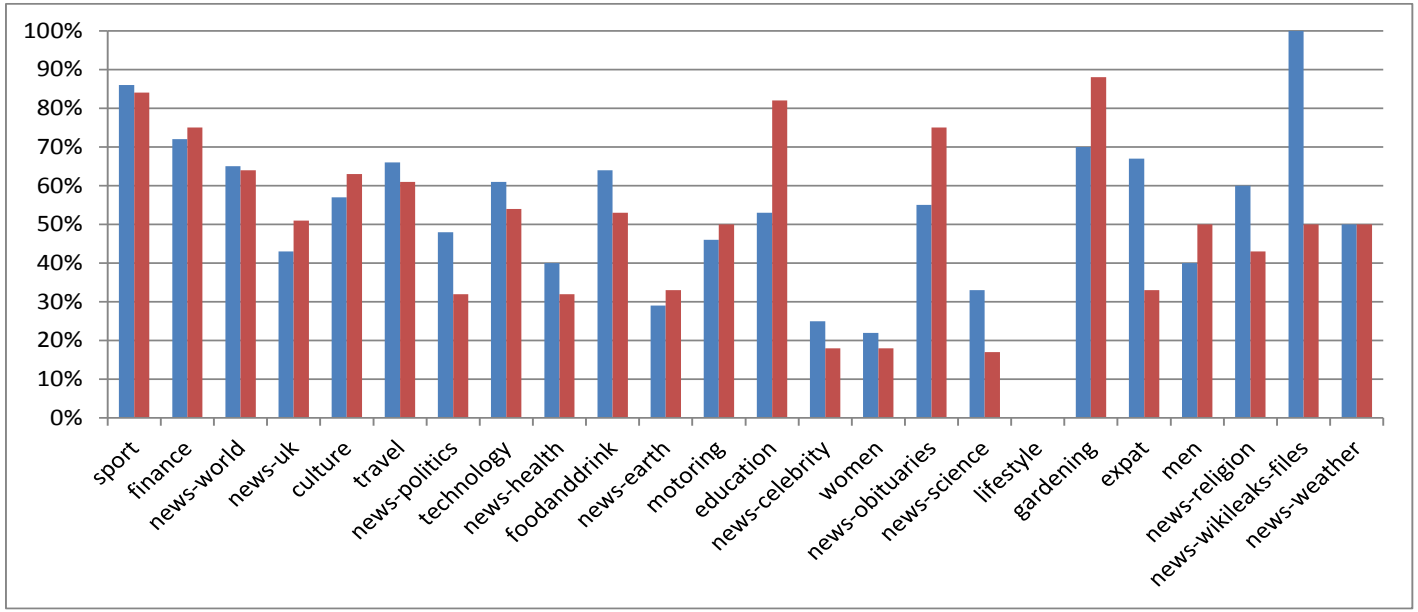
| | |
|---|---|
| All eyes on eclipse of the Moon | news-UK |
| All eyes on eclipse of the Moon | news-science |

2. **Category definition: Mutually exclusive, functionally relevant**

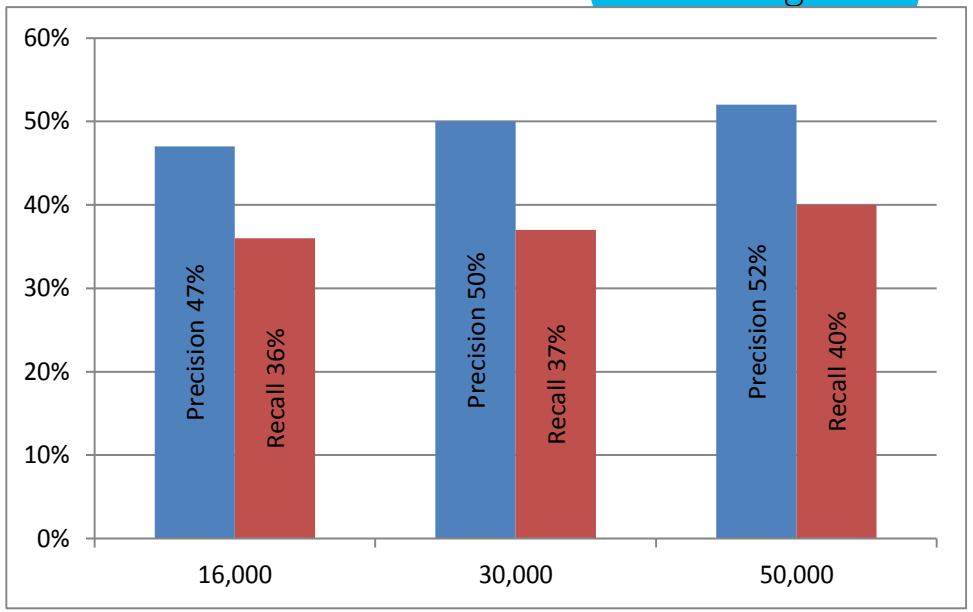
3. **Precision & Recall per categories and different training/test-sets**

4. **Eliminating Longtail**

abdulrahim kerimbakiev
abbot placid spearritt
abdullahi sudi arale
abdulah alhamiri
abib sarajuddin
acer nethercott
abdulli feghoul




Precision/recall per category




Precision/recall for different training/test- sizes

mgm

The result is **BETTER!**

Now…

… what **options** do you have if you don’t have a pre-categorized data set **to train your ML**?

… or your **manually classified data set is too small**?

mgm

# What to do about the things that can still go wrong

**Manually classified data set is too small for training**

- Data set is too heterogenous
- ML cannot detect patterns
- Bad precision and recall

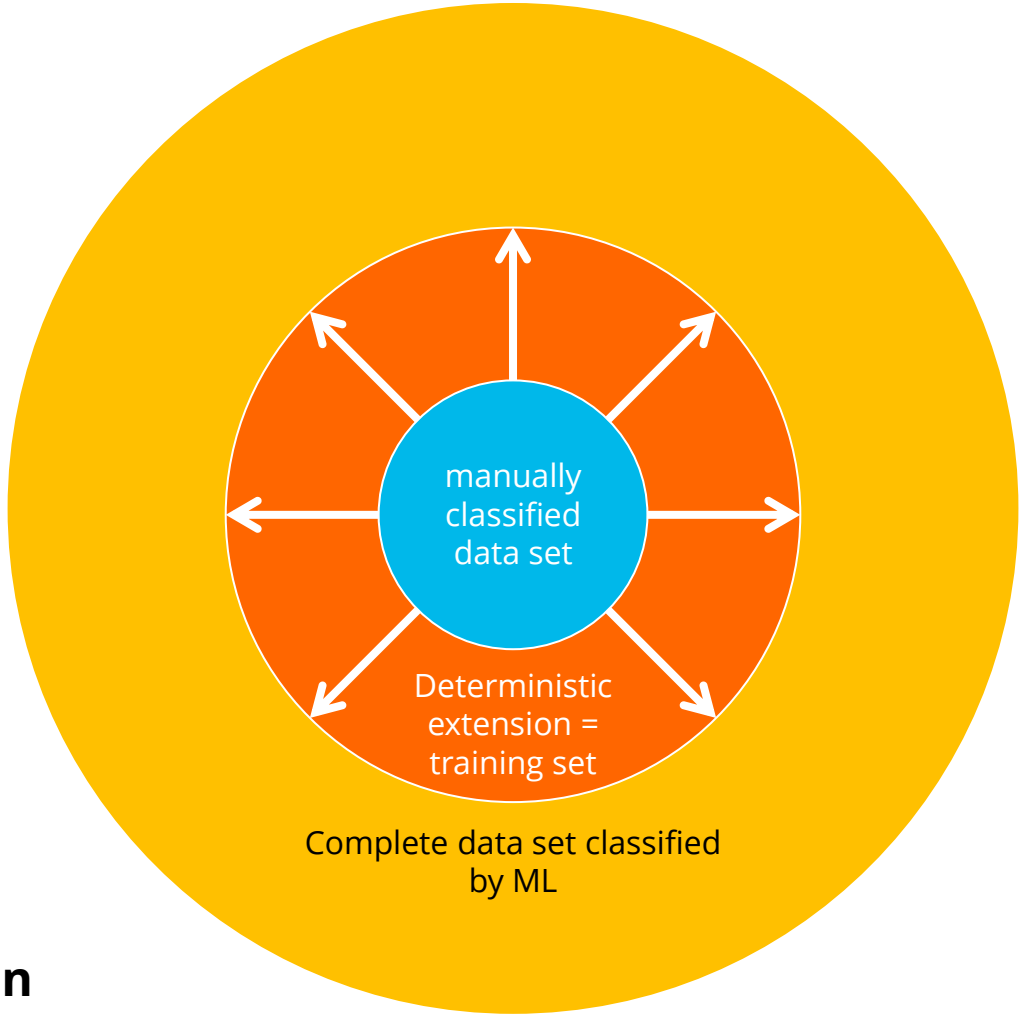
**Extend data set**

- Requires manual classification
- Too expensive

**Try to understand structure of manual classification**

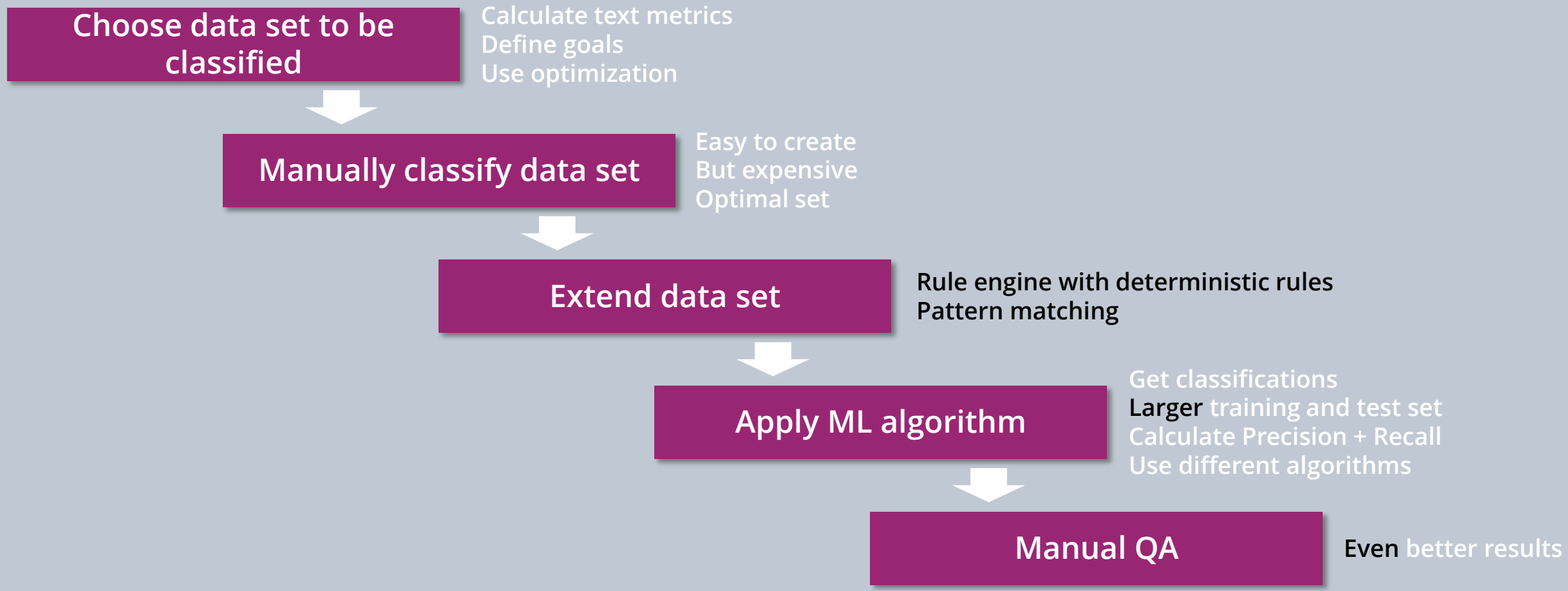
- Find category-specific keywords
- Find patterns
- Use NLP etc.

➔ **Extension of training set by deterministic classification**




mgm

# Improved approach with deterministic extension



# Be prepared for a long journey: Often results get better incrementally

**Input: Categories**

+ Data

**1 Find/adjust Categories**

| Task | Tools |
|---|---|
| Visualization | Solr & D3 |
| Clustering | Solr |
| Brain & Expertise | Data analytics |

**2 Documents to assign categories manually**

| Task | Tools |
|---|---|
| Manual classification | Mechanical turk |
| Deterministic classification | Rule engine |

**Iterate and Improve**

**3 Train and apply ML**

| Task | Tools |
|---|---|
| Training | R, Mahout, Spark |
| Test | R, Mahout, Spark |

**4 Measure results**

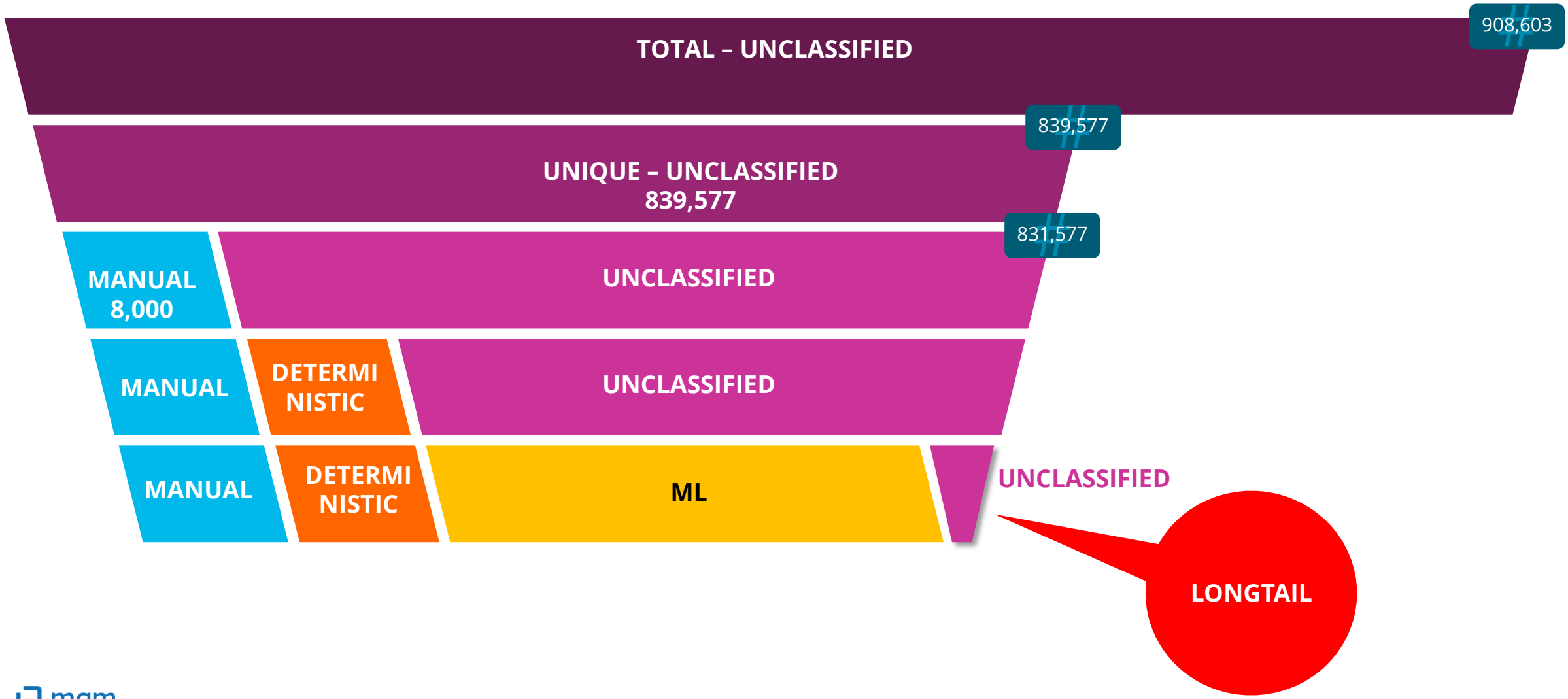
| Task | Tools |
|---|---|
| Check precision, recall, f-measure | R |
| | Apache Mahout |
| | Spark ML |

**Output: Categorized Data**

+ Histograms
+ Visualization
+ Metrics
+ Category specific keywords
+ Hierarchies, rules, entities

mgm

# Classification cascade




mgm

# Talking about longtail: Variability

**Reasons for longtail**

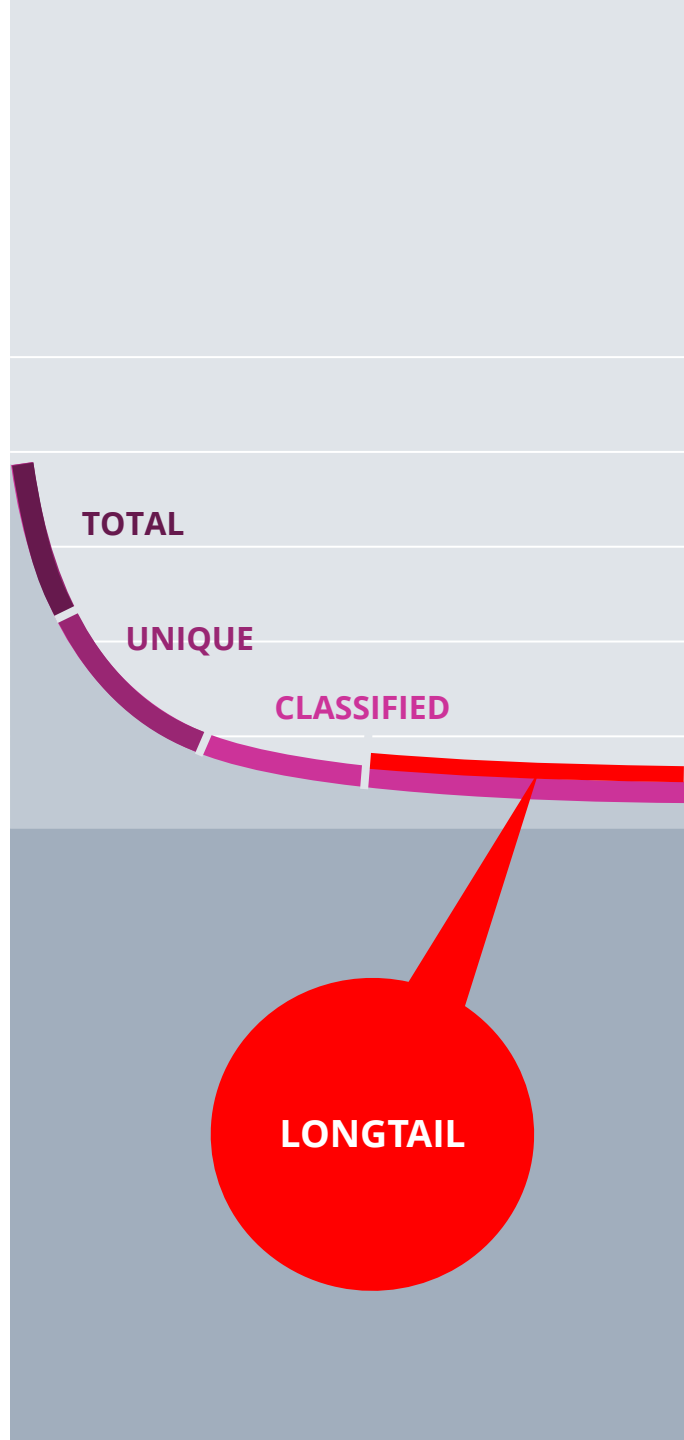
- Flat longtail via dictionary-type texts
- Decreasing longtail from domain specific language

**Analyze the longtail**

- Count words
- Measure heterogenity

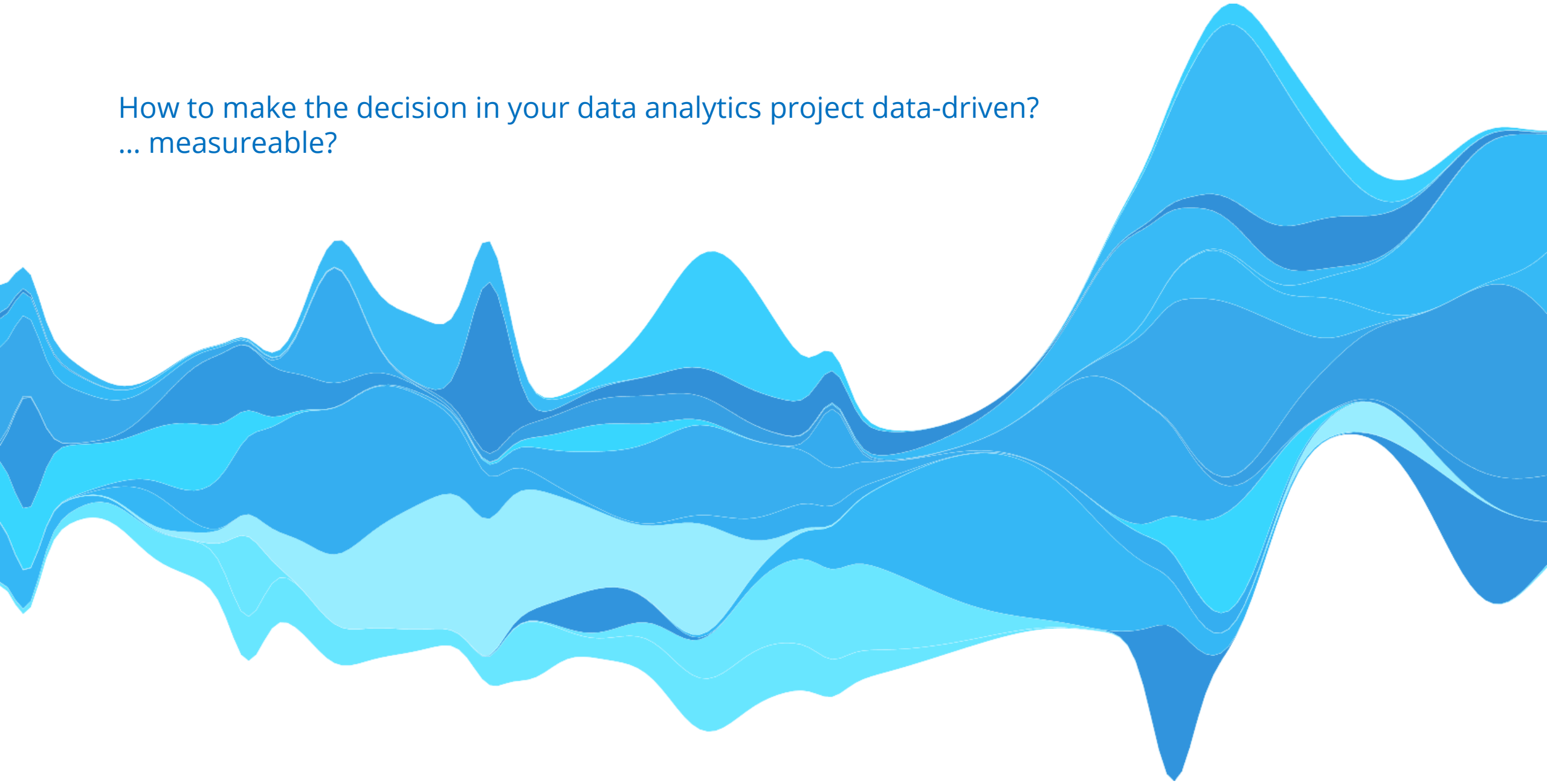
**Elimination strategies**

- Foreign language detection
- Eliminate typos (n-grams)
- Manual classification if not too many documents
- Put into separate category (aka „miscellaneous“)



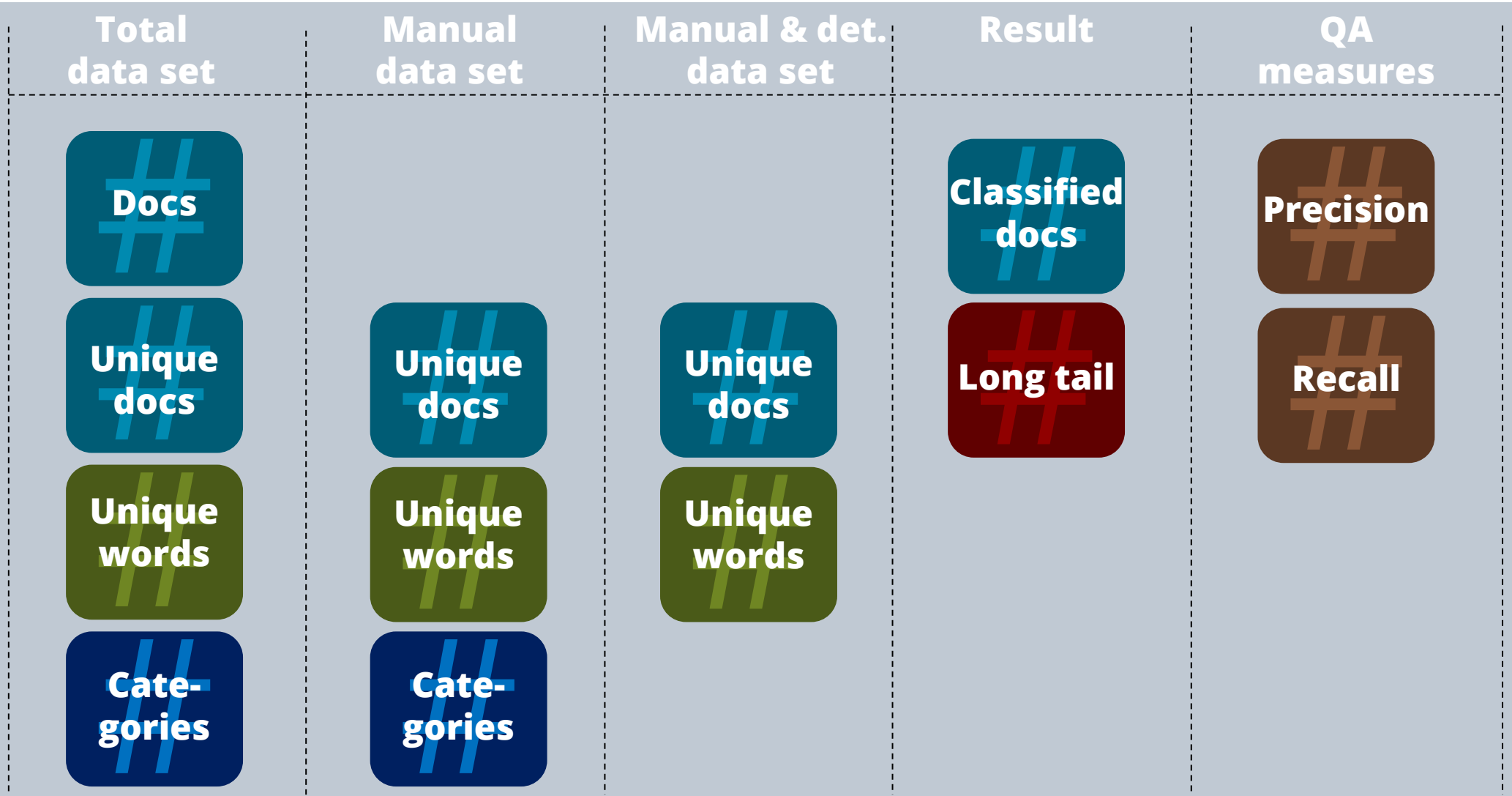

mgm

# How to make the decision in your data analytics project data-driven?
# … measureable?



mgm

# Metrics help making objective decisions during the project

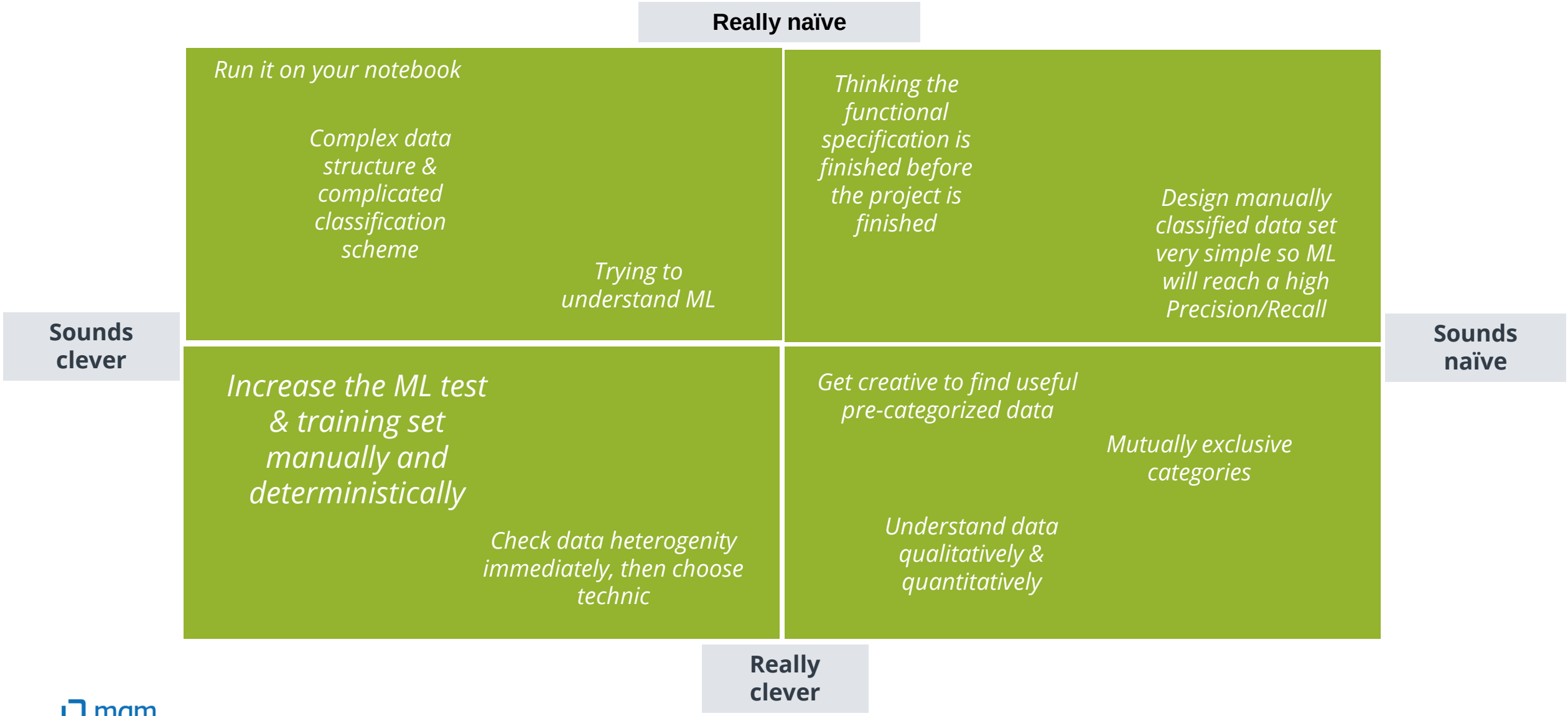



The project is finished when your cost/benefit ratio (or it's prediction) of classifying the longtail becomes negative.

05

# Conclusion and outlook

# 10 Lessons learned




mgm

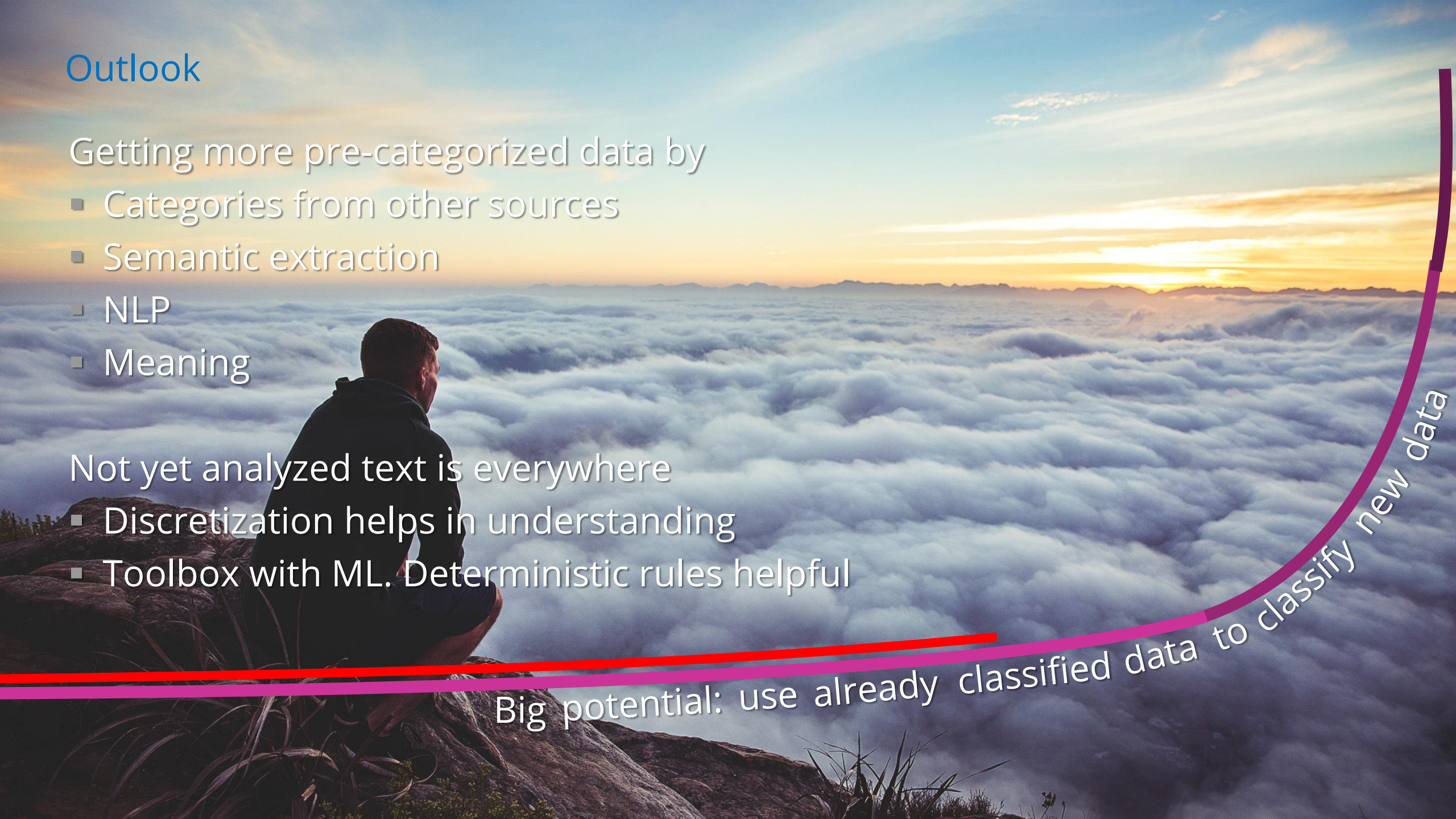
# Outlook

Getting more pre-categorized data by

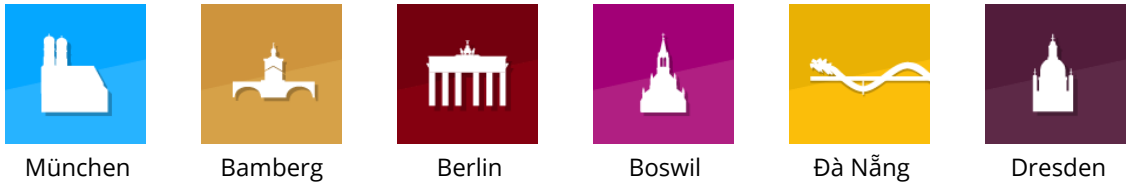
- Categories from other sources
- Semantic extraction
- NLP
- Meaning

Not yet analyzed text is everywhere

- Discretization helps in understanding
- Toolbox with ML. Deterministic rules helpful



# Innovation Implemented.



München | Bamberg | Berlin | Boswil | Đà Nẵng | Dresden

Grenoble | Hamburg | Köln | Leipzig | Nürnberg | Prag


**mgm technology partners GmbH**
Frankfurter Ring 105a
80807 München
Tel.: +49 (89) 35 86 80-0
Fax: +49 (89) 35 86 80-288
http://www.mgm-tp.com

**mgm consulting partners GmbH**
Holländischer Brook 2
20457 Hamburg
Tel.: +49 (0) 40 / 80 81 28 20-0
Fax: +49 (0) 40 / 80 81 28 20-388
http://www.mgm-cp.com


mgm