Imperial College London

June 9 2021

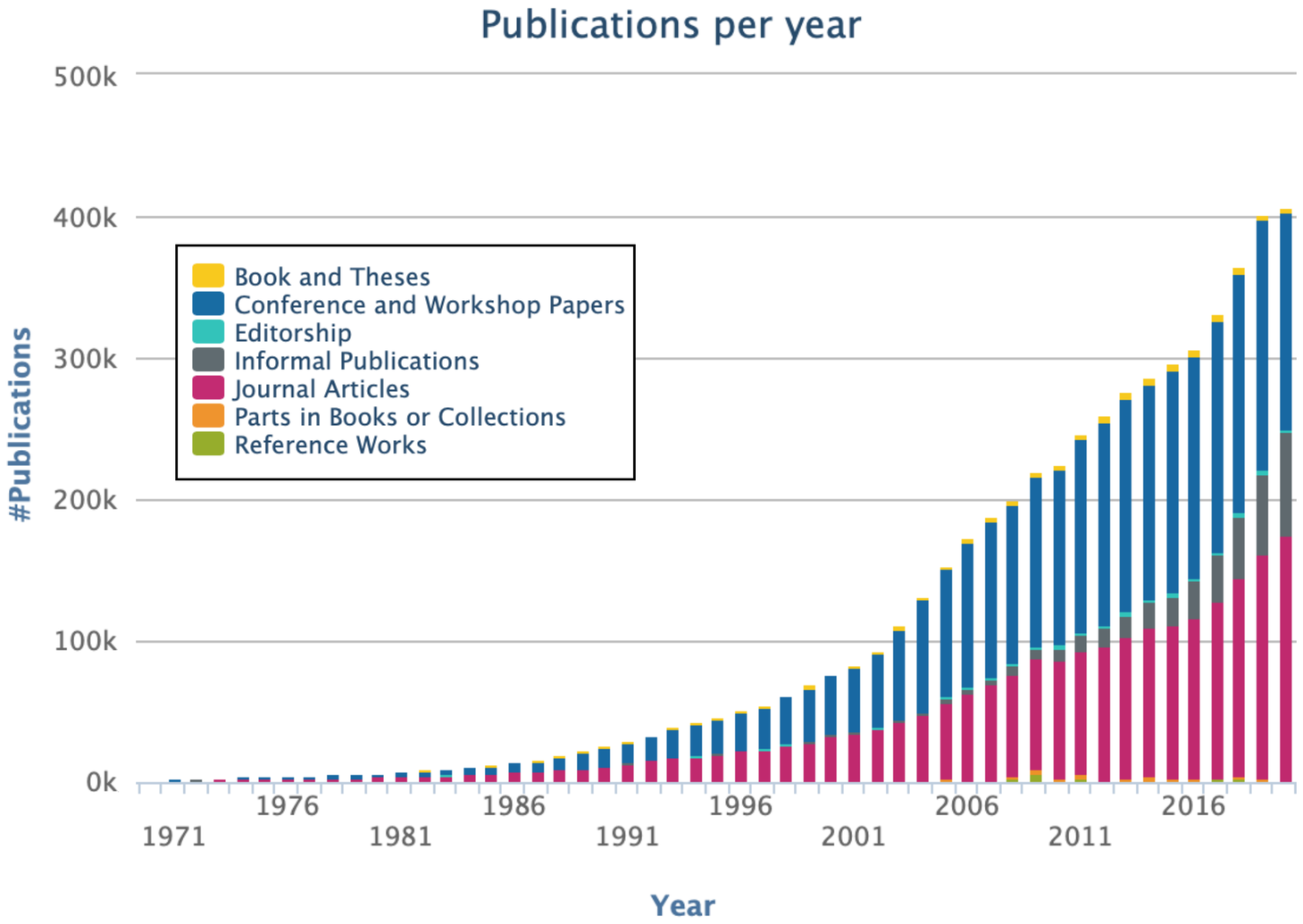
# Novelty Detection in Scientific Research Papers

Yousef Nami

Supervisor: Dr Loïc Salles

# Contents

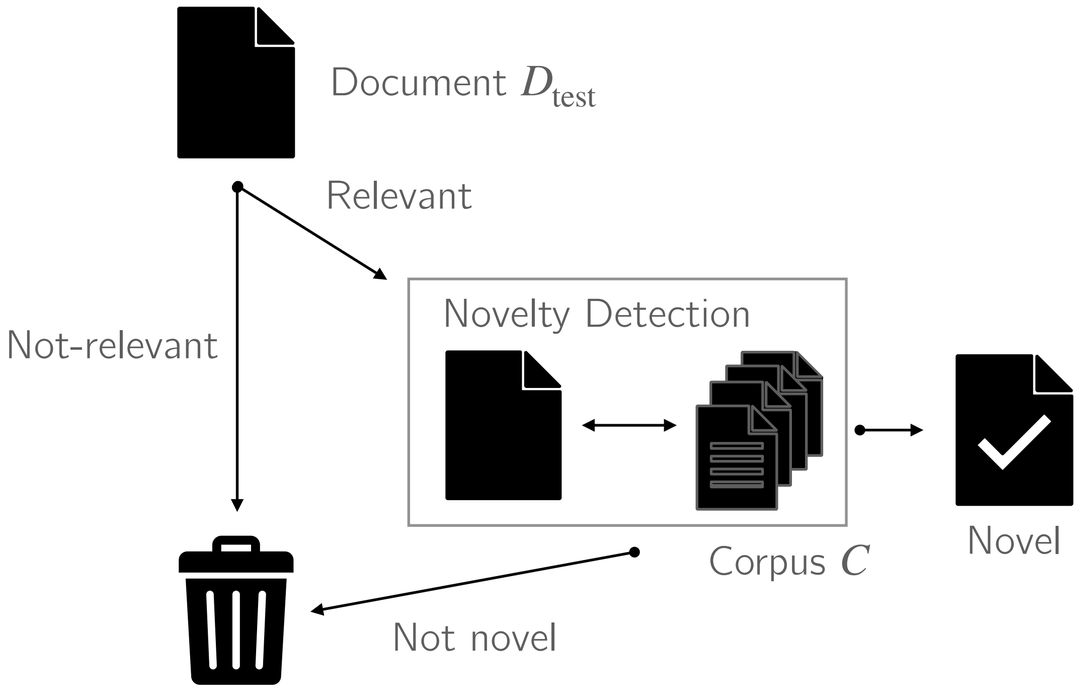
# Problem Definition




[1] *Publications per year.* Digital Bibliography and Library project. Link: dblp.org/statistics/publicationsperyear.html

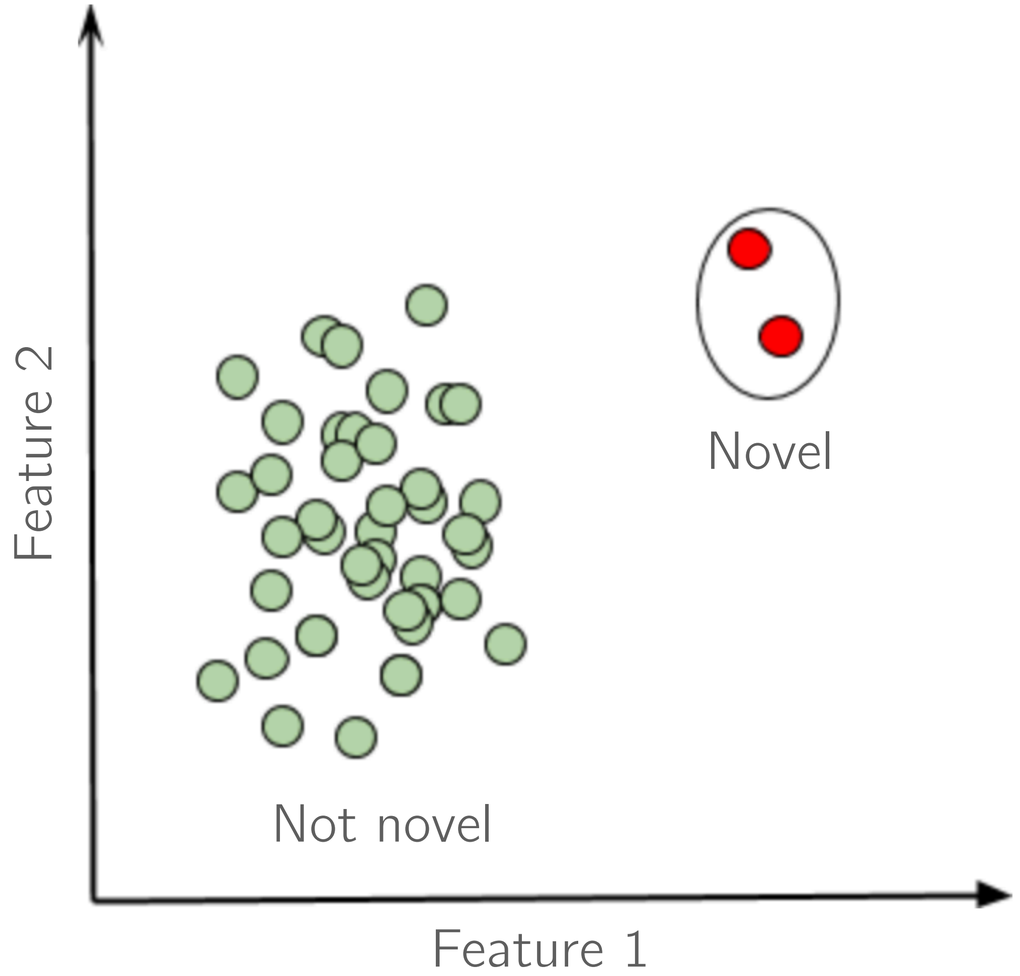
# Problem Definition

- Need for tools that distinguish novel subject matter from redundant content
- **Novelty:** dissimilarity provided that a document is relevant
- Needs deep semantic representation of documents



# Problem Definition

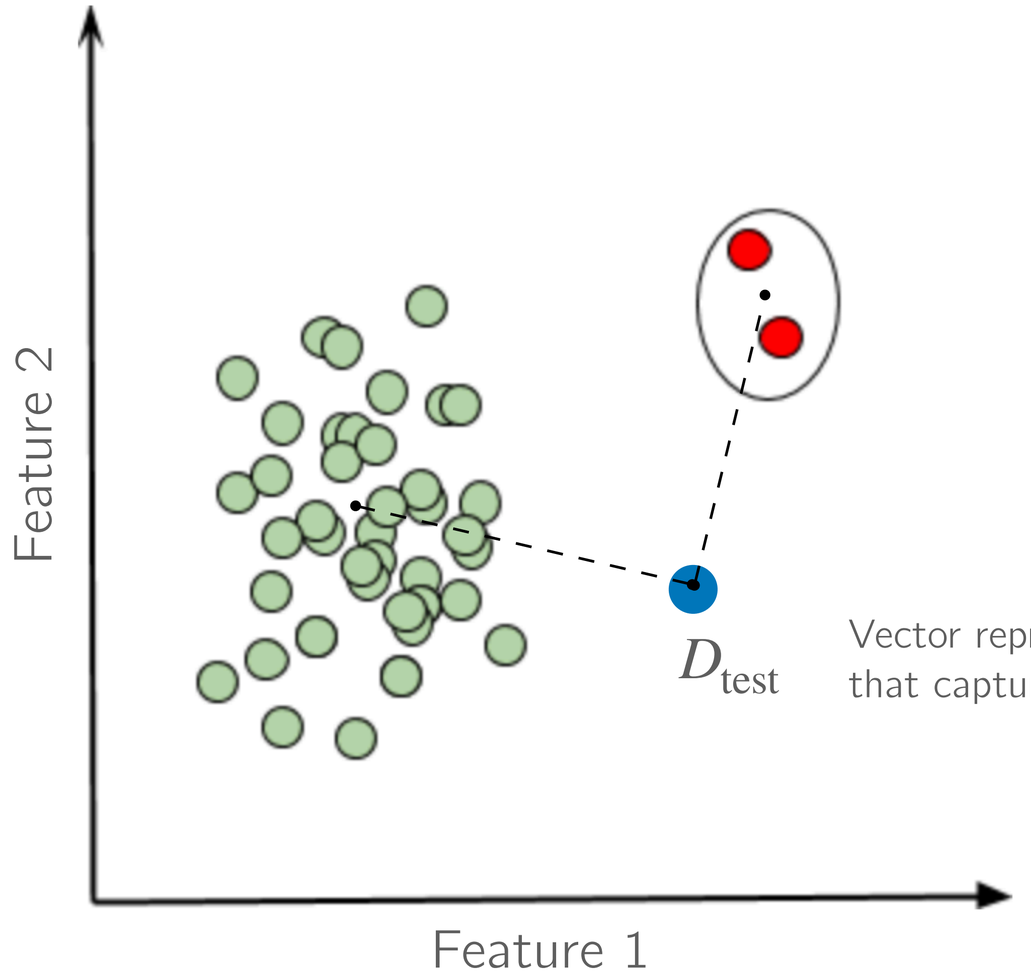
**Novelty**
Dissimilarity provided that a document is relevant




[2] *Introduction to Anomaly Detection.* FloydHub. Link: blog.floydhub.com/introduction-to-anomaly-detection-in-python/

# Problem Definition

**Novelty**
Dissimilarity provided that a document is relevant



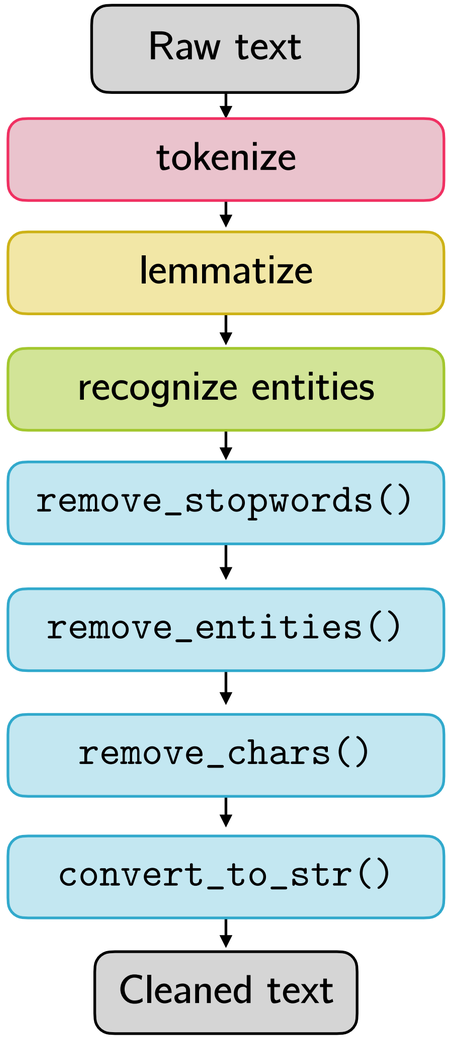

Vector representation of a document that captures it's semantic meaning

[2] *Introduction to Anomaly Detection.* FloydHub. Link: blog.floydhub.com/introduction-to-anomaly-detection-in-python/

# Data Sources

| | Details | Extraction Method |
|---|---|---|
| **Training** | **ASME Turbomachinery**<br>80% of 26030 PDF files | ▸ Data stored on remote linux device<br>▸ Data shuffled `train_test_split` from `sklearn`<br>▸ PDFs converted to text using OCR library `pdfminer` |
| **Validation** | Complementing 20% of above files | |
| **Testing** | **Hindawi**<br>*Shock and Vibration* and *International Journal of Rotating Machines* (2008—2020) stored in XML format | ▸ Custom XML parser using `xml.etree.ElementTree` from Python |
| | **IEEE**<br>Papers on "Deep Learning" from Open Access collection stored in JSON format | ▸ User query extraction using publisher API |
| | **Core**<br>Papers on "Turbomachinery" stored in JSON format | |

# Data Cleaning




*Where you stand depends on where you sit - Nelson Mandela*

```
["Where", "you", "stand", "depends", "on", "where", "you", "sit", "-", "Nelson", "Mandela"]
```

depends —> depend

PERSON

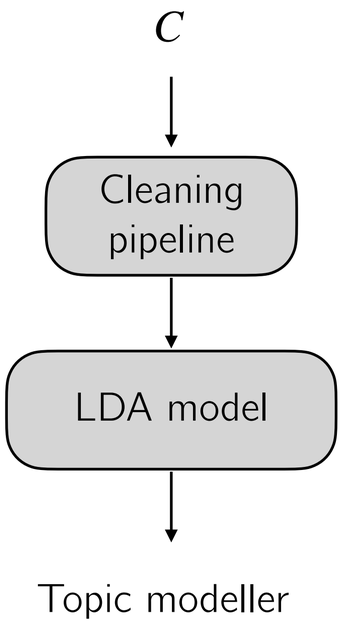
```
["stand","depends", "sit", "-", "Nelson", "Mandela"]
["stand","depends", "sit", "-"]
["stand","depends", "sit"]
["stand","depend", "sit"]
```

*stand depend sit*

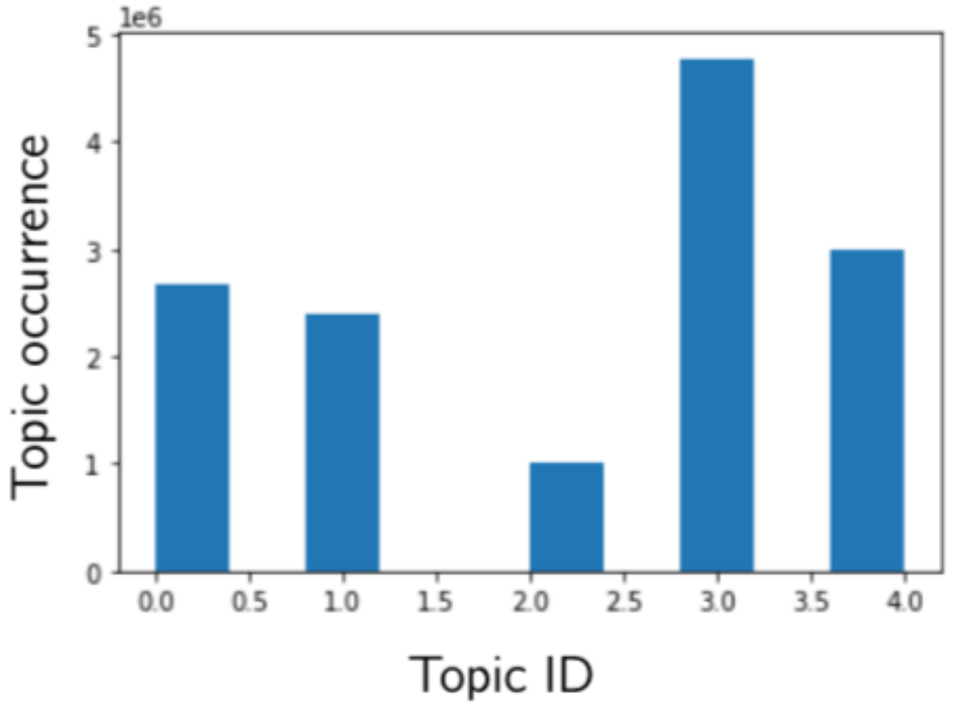
- tokenize: separate text into smaller chunks
- lemmatize: get root word
- recognize entities: recognize if word is a known entity
- custom functions: for cleaning

# Exploratory Data Analysis

- Statistical model known as Latent Dirichlet Allocation (LDA)
- Learns the topic distribution of documents
- Effectively finds the probability that a word $w_i$ belongs to any of $m$ topics selected by user

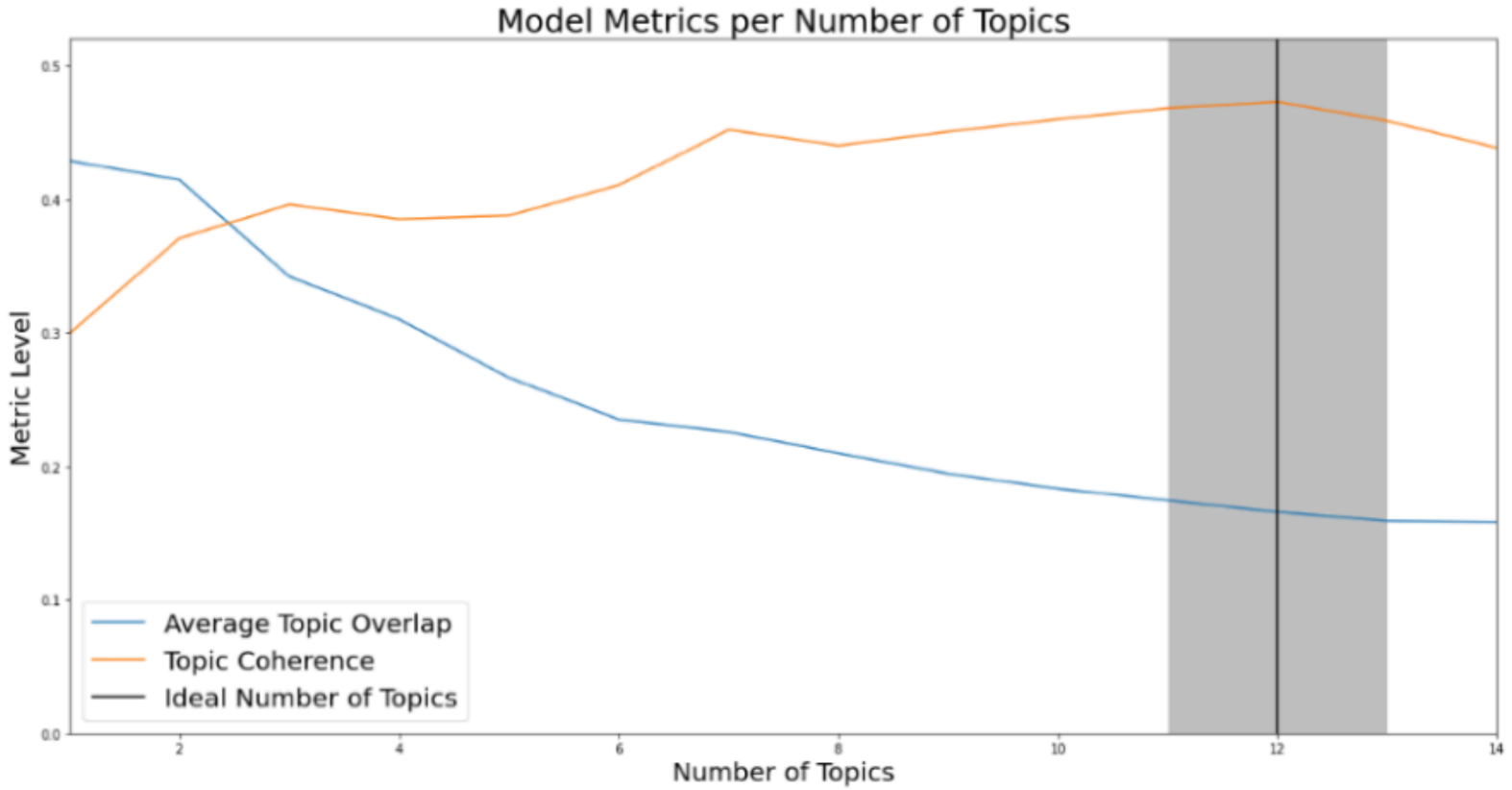


# Exploratory Data Analysis




```
[(0,
  '0.016*"flow" + 0.016*"flow" + 0.010*"case" + 0.009*"pressure" + 0.009*"high" + 0.009
*"figure" + 0.008*"show" + 0.008*"wall" + 0.008*"surface" + 0.007*"result"'),
 (1,
  '0.022*"blade" + 0.020*"rotor" + 0.018*"flow" + 0.015*"pressure" + 0.011*"design" +
0.011*"figure" + 0.010*"compressor" + 0.008*"show" + 0.008*"stage" + 0.008*"speed"'),
 (2,
  '0.015*"fuel" + 0.013*"combustor" + 0.013*"combustion" + 0.012*"temperature" + 0.012
*"flame" + 0.011*"air" + 0.009*"flame" + 0.009*"pressure" + 0.009*"high" + 0.009*"veloc
ity"'),
 (3,
  '0.015*"model" + 0.008*"method" + 0.008*"figure" + 0.006*"result" + 0.006*"time" + 0
.006*"value" + 0.006*"base" + 0.005*"analysis" + 0.005*"show" + 0.005*"system"'),
 (4,
  '0.016*"temperature" + 0.013*"turbine" + 0.012*"engine" + 0.012*"system" + 0.011*"po
wer" + 0.010*"pressure" + 0.010*"design" + 0.009*"high" + 0.008*"figure" + 0.008*"perf
ormance"')]
```

# Exploratory Data Analysis




Jaccard score

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

**Coherence**

Measures the semantic similarity

# Text as vectors

## Bag-of-words

- Consider $C = \{\, "My\ name\ is\ Yousef",\ "NLP\ is\ awesome" \}$
- Represented by matrix $V_{ij} \in \mathbb{R}^{N \times L}$ where $N$ is # of documents, $L$ size of unique vocabulary
- Does not capture deep semantic information!

| | $My$ | $name$ | $is$ | $Yousef$ | $NLP$ | $awesome$ |
|---|---|---|---|---|---|---|
| $My\ name\ is\ Yousef$ | 1 | 1 | 1 | 1 | 0 | 0 |
| $NLP\ is\ awesome$ | 0 | 0 | 1 | 0 | 1 | 1 |

# Text as vectors

## TF-IDF

- Product of **term frequency** and **inverse document frequency**
- Punishes words that appear in all documents
- Still fails to capture deep semantic information!

$$\text{tf}_{ij} = \frac{\#\text{ occurances of } w_j \text{ in } D_i}{\#\text{ unique tokens in } D_i} = P(w_j | D_i)$$

$$\text{idf}_i = \frac{\#\text{ documents in corpus}}{\#\text{ documents containing word } i}$$

$$V_{ij} = \text{tf}_{ij} \odot \text{idf}_i$$

| | $My$ | $name$ | $is$ | $Yousef$ | $NLP$ | $awesome$ |
|---|---|---|---|---|---|---|
| $My\ name\ is\ Yousef$ | 0.5 | 0.5 | **0.25** | 0.5 | 0 | 0 |
| $NLP\ is\ awesome$ | 0 | 0 | **0.33** | 0 | 0.67 | 0.67 |

# Text as vectors

## GloVe

- Represent each word $w_i$ as a vector $\underline{v} \in \mathbb{R}^l$, where $l \in [50, 300]$
- Word vectors carry semantic meaning
- Additive meaning

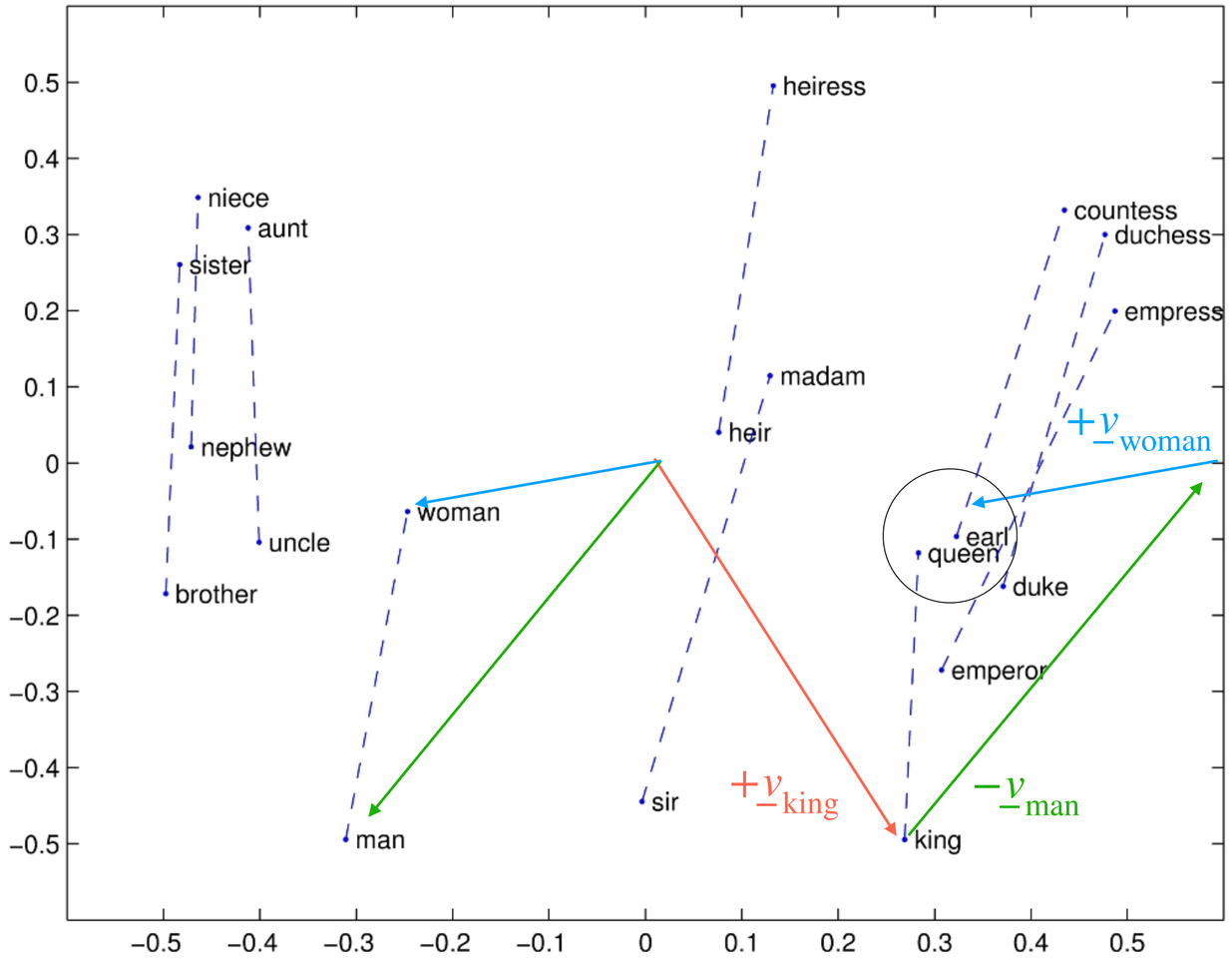
$$\underline{v}_{woman} + \underline{v}_{man} - \underline{v}_{queen} \approx \underline{v}_{king}$$

- Spatial closeness

$$\underline{v}_{Apple} \cdot \underline{v}_{IBM} \approx 1$$

# Text as vectors

## GloVe




**Additive meaning**

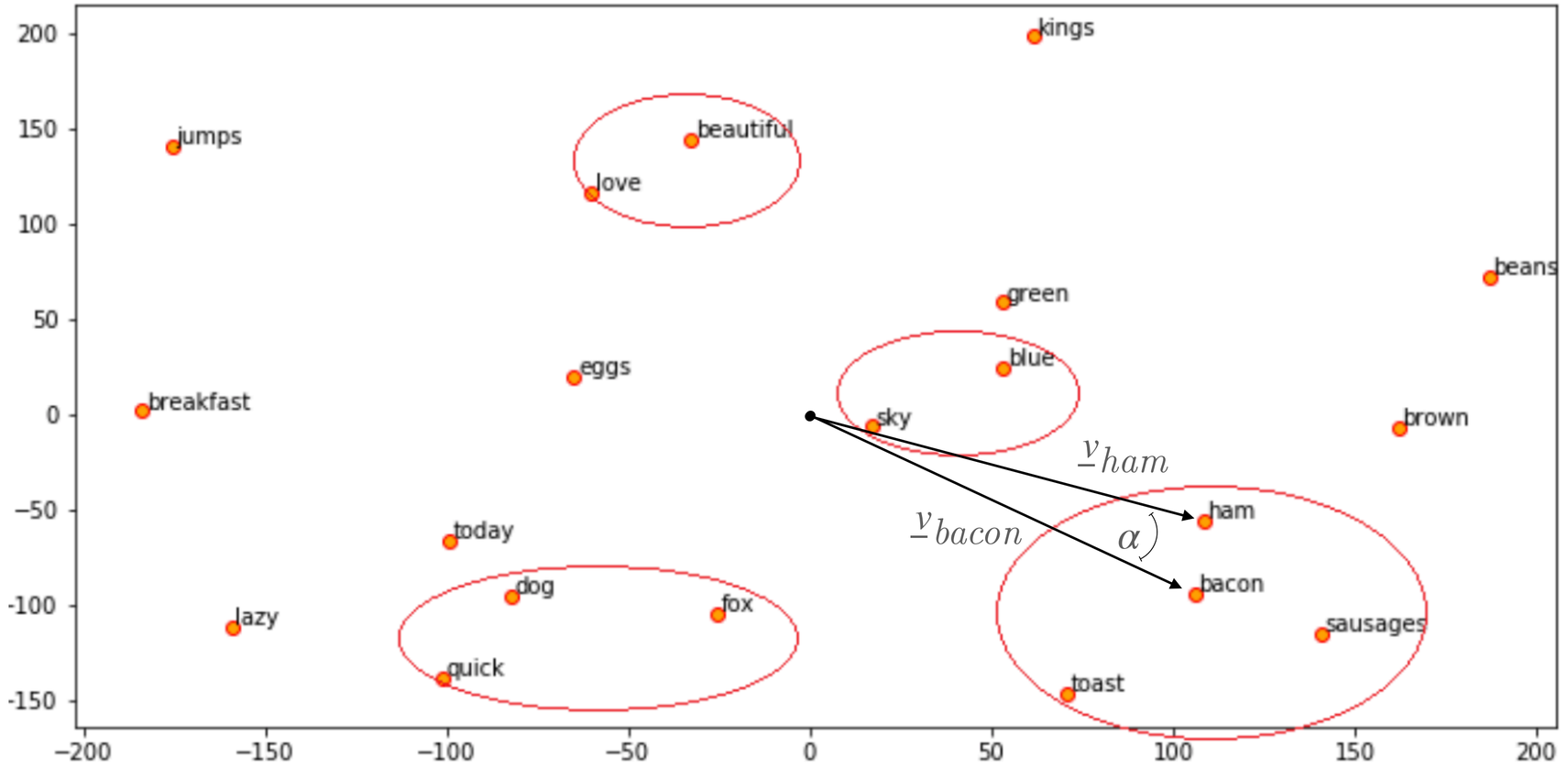
Vectors can add to each other, mimicking real relationships

$$\underline{v}_{king} - \underline{v}_{man} + \underline{v}_{woman} \approx \underline{v}_{queen}$$

[3] *GloVe: Global Vectors for Word Representation.* Pennington et al. Link: nlp.stanford.edu/projects/glove/

# Text as vectors

## GloVe




**Spatial closeness**

clusters indicate semantic family

$C = \{"quick",\ "dog",\ "fox"\} \approx "animal"$

$$\cos\alpha = \underline{v}_{ham} \cdot \underline{v}_{bacon} \approx 1$$

[4] *Hands on approach to Deep Learning methods for text data.* Dipanjan Sarkar. Link: towardsdatascience.com/understanding-feature-engineering-part-4-deep-learning-methods-for-text-data-96c44370bbfa

# Text as vectors

## GloVe

- GloVe vectors found by matrix factorisation of $F(w_i, w_j, \tilde{w}_k) = P_{ik}/P_{jk}$

| Probability and Ratio | $k = solid$ | $k = gas$ | $k = water$ | $k = fashion$ |
|---|---|---|---|---|
| $P(k\|ice)$ | $1.9 \times 10^{-4}$ | $6.6 \times 10^{-5}$ | $3.0 \times 10^{-3}$ | $1.7 \times 10^{-5}$ |
| $P(k\|steam)$ | $2.2 \times 10^{-5}$ | $7.8 \times 10^{-4}$ | $2.2 \times 10^{-3}$ | $1.8 \times 10^{-5}$ |
| $P(k\|ice)/P(k\|steam)$ | 8.9 | $8.5 \times 10^{-2}$ | 1.36 | 0.96 |

Very small or large:

solid is related to ice but not steam, or
gas is related to steam but not ice

close to 1:

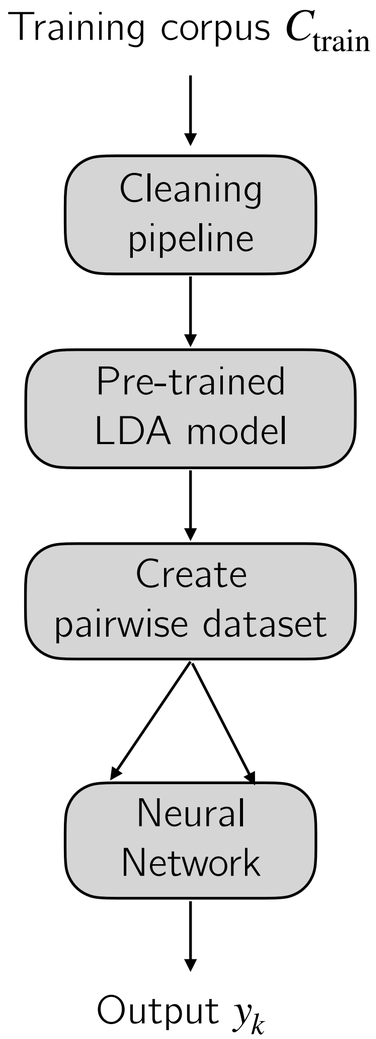
water is highly related to ice and steam, or
fashion is not related to ice or steam.

[5] *Word Embedding and GloVe.* Jonathan Hui. Link: https://jonathan-hui.medium.com/nlp-word-embedding-glove-5e7f523999f6

# Pseudo-Supervised Model

## Training process

- Assigns a topic to each sentence
- Pairs all sentences together
- Trains a neural network to predict the probability that a pair belongs in the same class
- Model based on pairwise network by Qin et al. [6]




[6] *Text Classification with novelty detection.* Qin Q, Hu W, Liu B. Link: https://arxiv.org/abs/2009.11119

# Pseudo-Supervised Model

## Pairwise data

| | Topic ID |
|---|---|
| *My name is Yousef* | 0 |
| *NLP is awesome* | 1 |
| *Deep Learning* | 1 |

| Topic ID | Meaning |
|---|---|
| 0 | People |
| 1 | Artificial Intelligence |

Sentence pair

| Sent 1 | Sent 2 | Topic ID 1 | Topic ID 2 | Same topic? |
|---|---|---|---|---|
| *My name is Yousef* | *NLP is awesome* | 0 | 1 | 0 |
| *My name is Yousef* | *Deep Learning* | 0 | 1 | 0 |
| *NLP is awesome* | *Deep Learning* | 1 | 1 | 1 |

# Pseudo-Supervised Model

## Testing process

- Topic 0: People
- Topic 1: Artificial Intelligence

| Test sentence | Sentences from corpus | Probability | Mean by topic |
|---|---|---|---|
| *Machine learning is very broad* | *My name is Yousef* | 0.1 | 0.1 |
| *Machine learning is very broad* | *NLP is awesome* | 0.7 | **0.75** |
| *Machine learning is very broad* | *Deep Learning* | 0.8 | |

Not-novel

| Test sentence | Sentences from corpus | Probability | Mean by topic |
|---|---|---|---|
| *A turbine is a Turbomachine* | *My name is Yousef* | 0.1 | 0.1 |
| *A turbine is a Turbomachine* | *NLP is awesome* | 0.3 | 0.25 |
| *A turbine is a Turbomachine* | *Deep Learning* | 0.2 | |

Novel

# Unsupervised Model

- Document representation found using mean GloVe word vectors

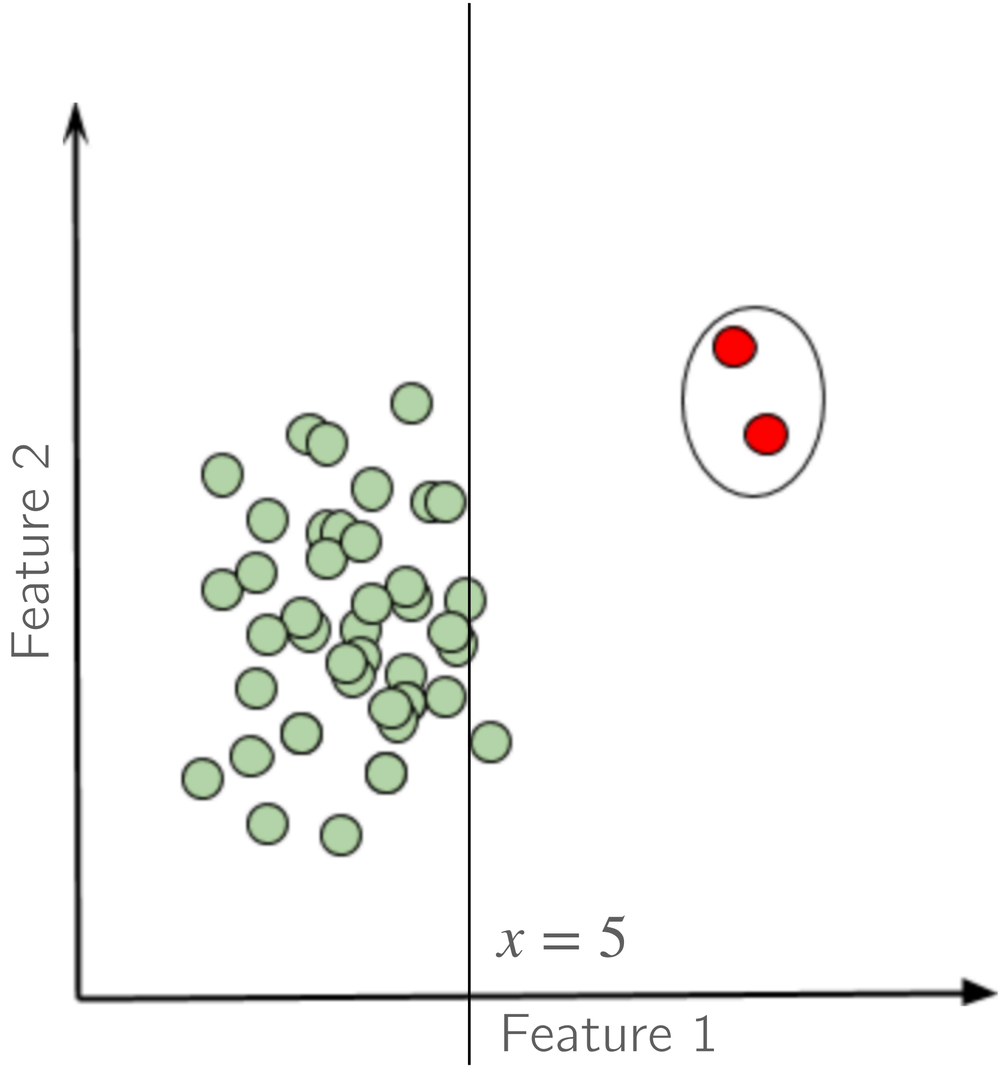
$$\bar{v} = \frac{\sum_i^N v_i}{N}$$

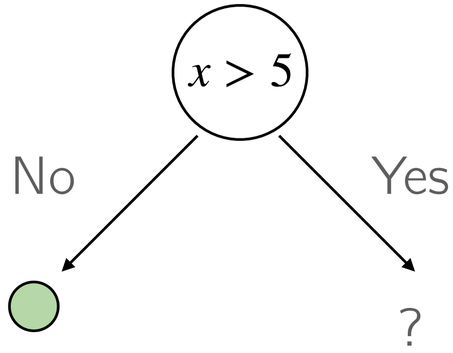
$N$ is the # of words per document

$v_i$ is the word vector for word $i$

- IsolationForest algorithm was used to determine anomalies
- Results compared with TF-IDF embedded documents

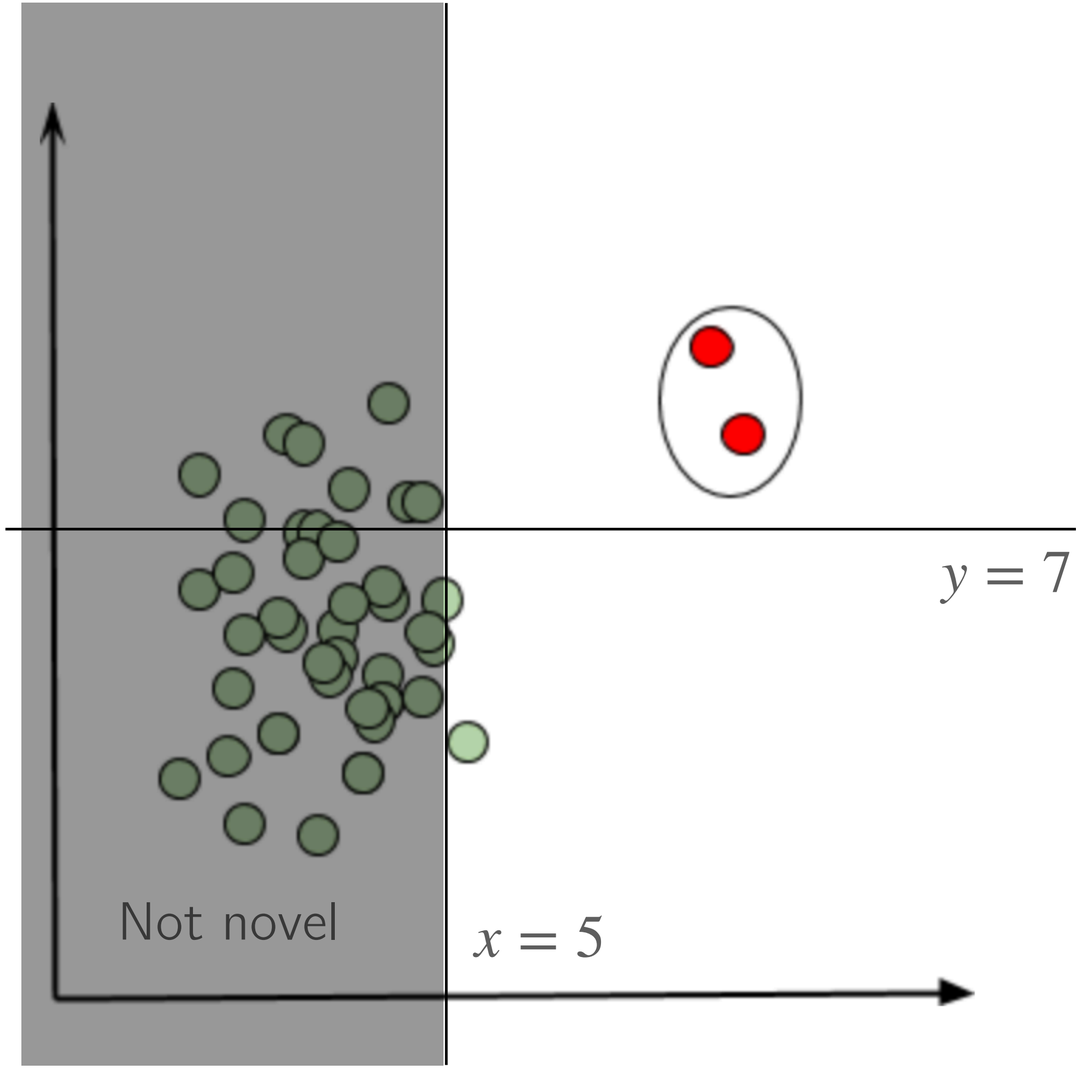
# Unsupervised Model

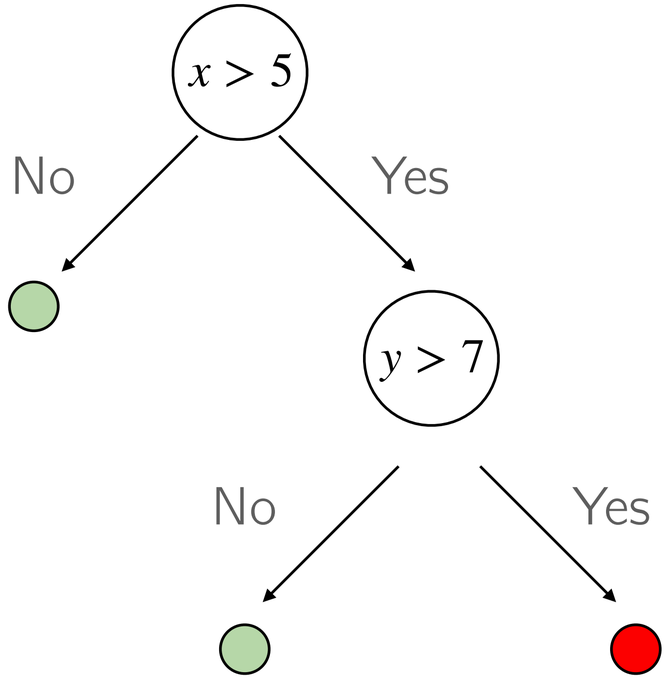



[2] *Introduction to Anomaly Detection.* FloydHub. Link: blog.floydhub.com/introduction-to-anomaly-detection-in-python/

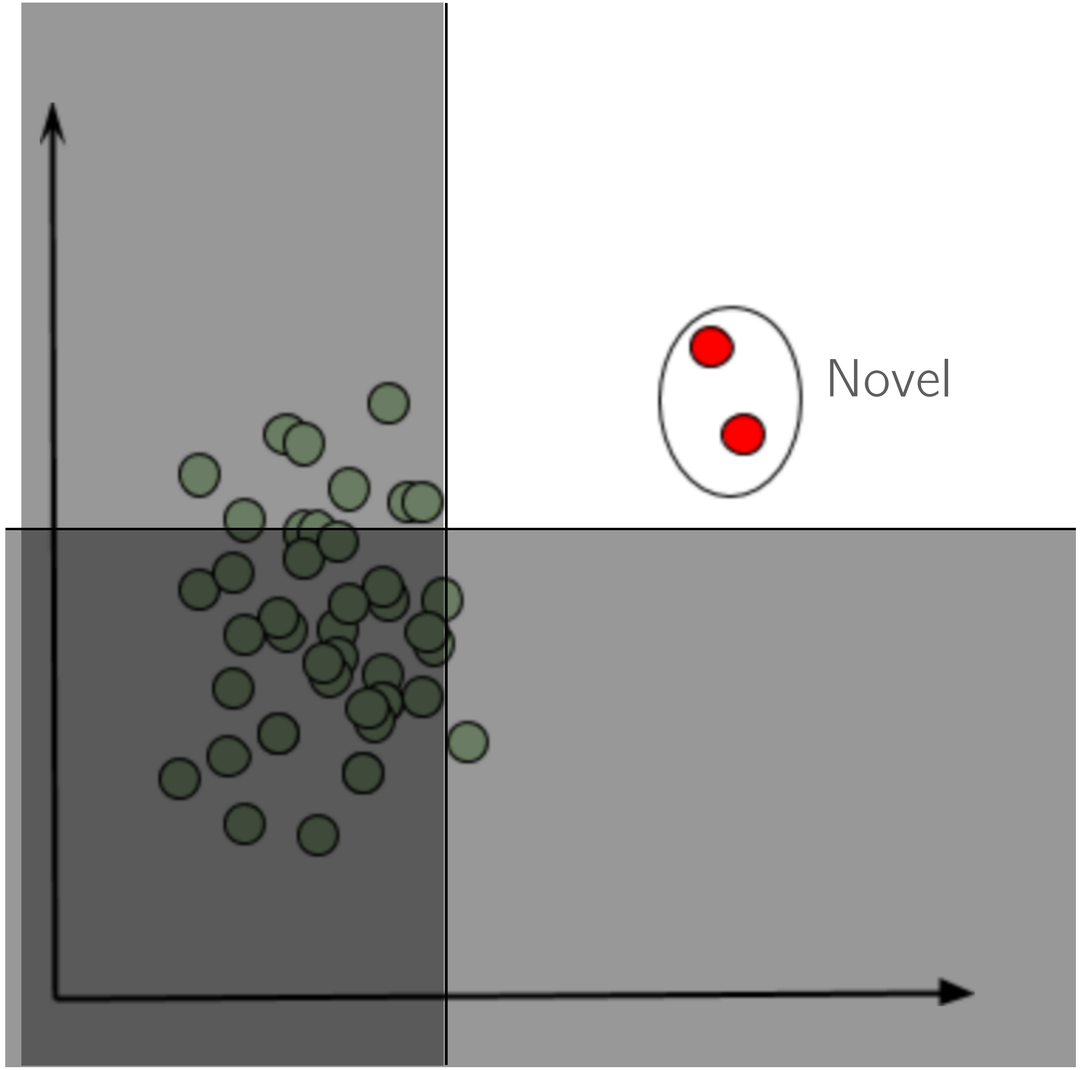
# Unsupervised Model

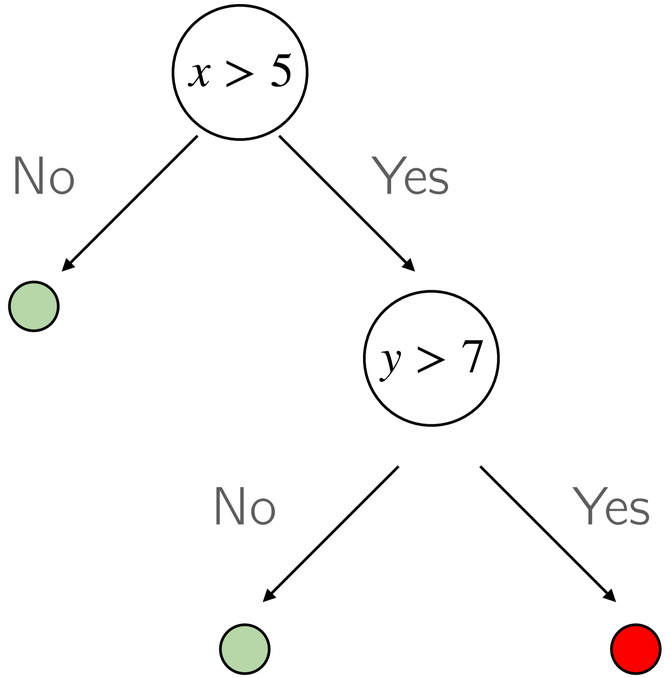



[2] *Introduction to Anomaly Detection.* FloydHub. Link: blog.floydhub.com/introduction-to-anomaly-detection-in-python/

# Unsupervised Model

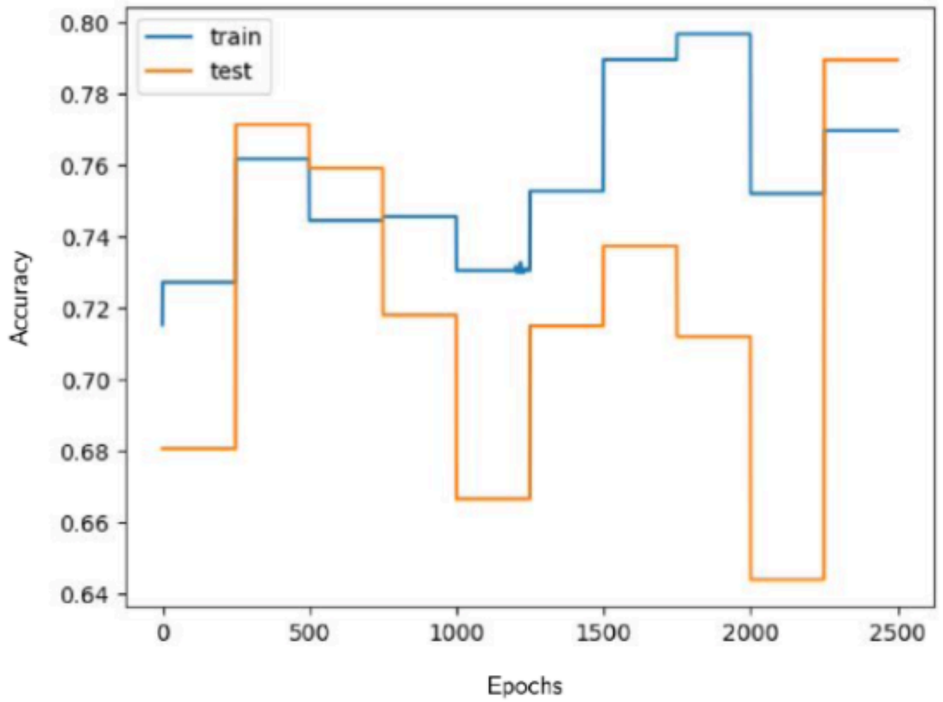


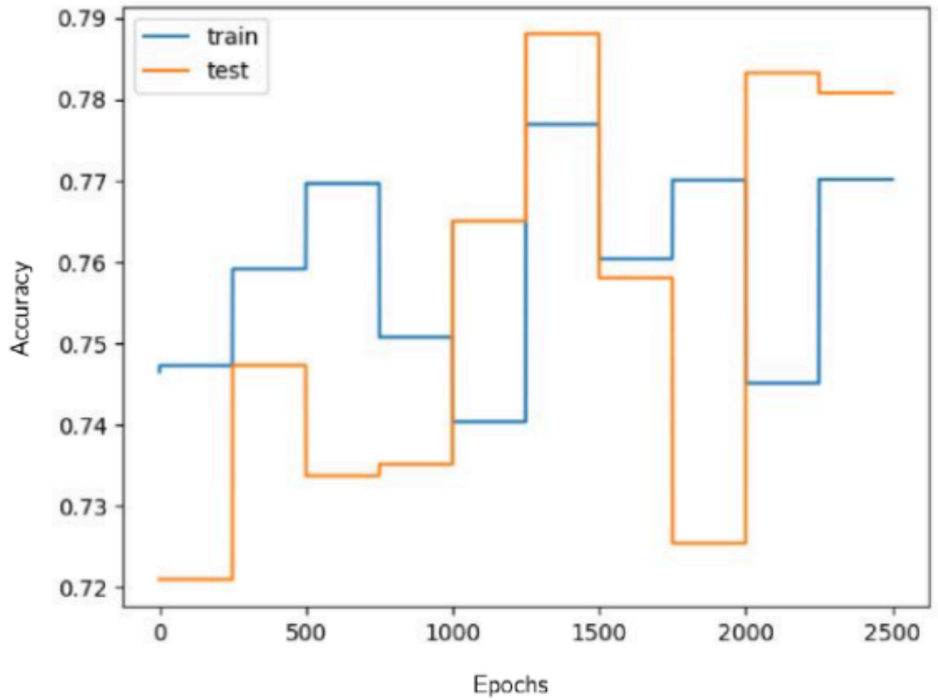

[2] *Introduction to Anomaly Detection.* FloydHub. Link: blog.floydhub.com/introduction-to-anomaly-detection-in-python/

# Project Outcomes and Discussion

## Pseudo-Supervised Model




$n = 200$ sentences

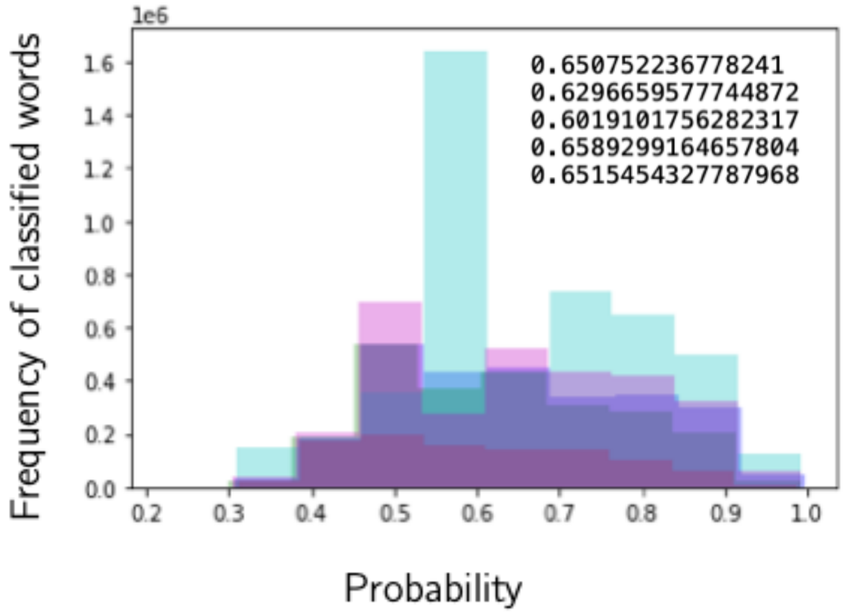



$n = 1000$ sentences

# Project Outcomes and Discussion

## Pseudo-Supervised Model

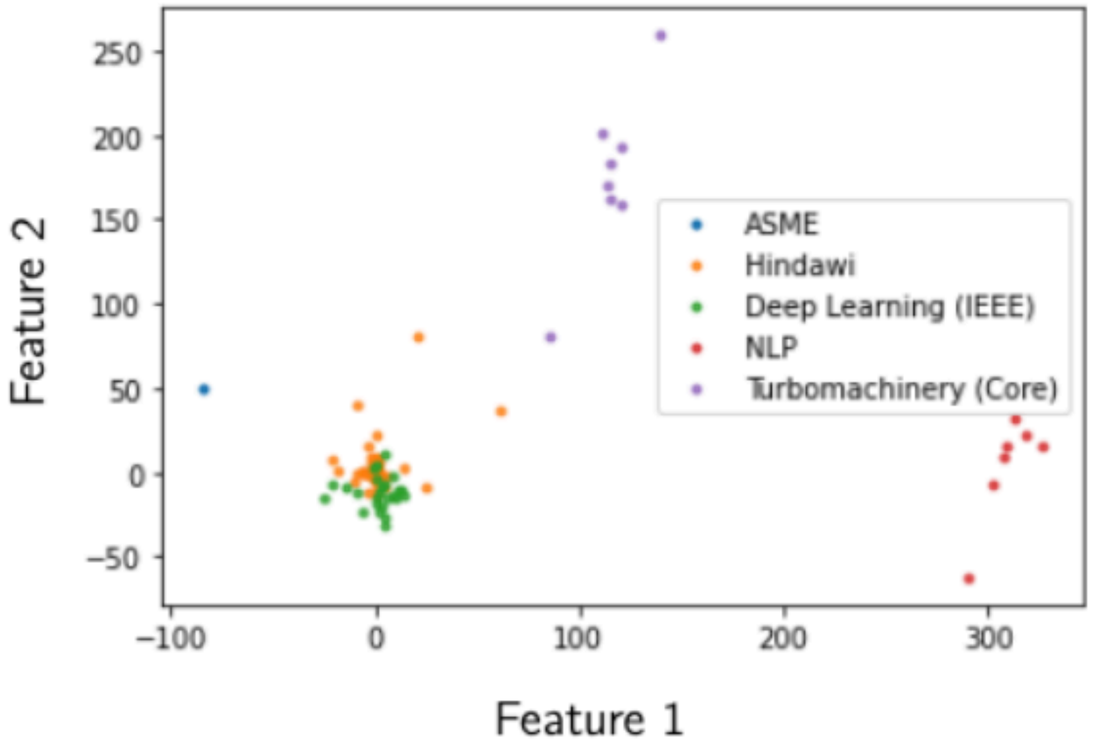
- Pseudo-supervised nature of the model not rigorously analysed
- Assumed that the LDA model can generate topics both at corpus (macro) and sentence (micro) level
- Degree of prediction confidence not considered
- Should set a threshold (e.g. 0.8)
- Would decrease computational cost

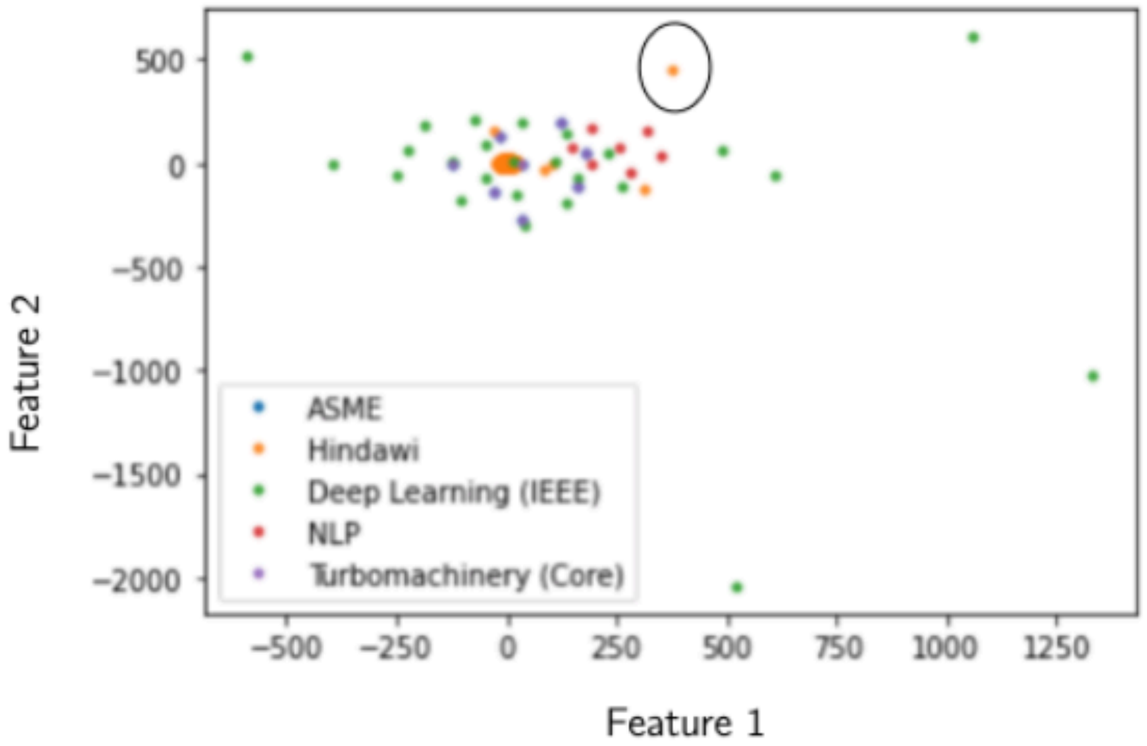


# Project Outcomes and Discussion

## Unsupervised Model

$l = L$ length document vectors projected in 2D plane where $L$ is the unique number of words in corpus

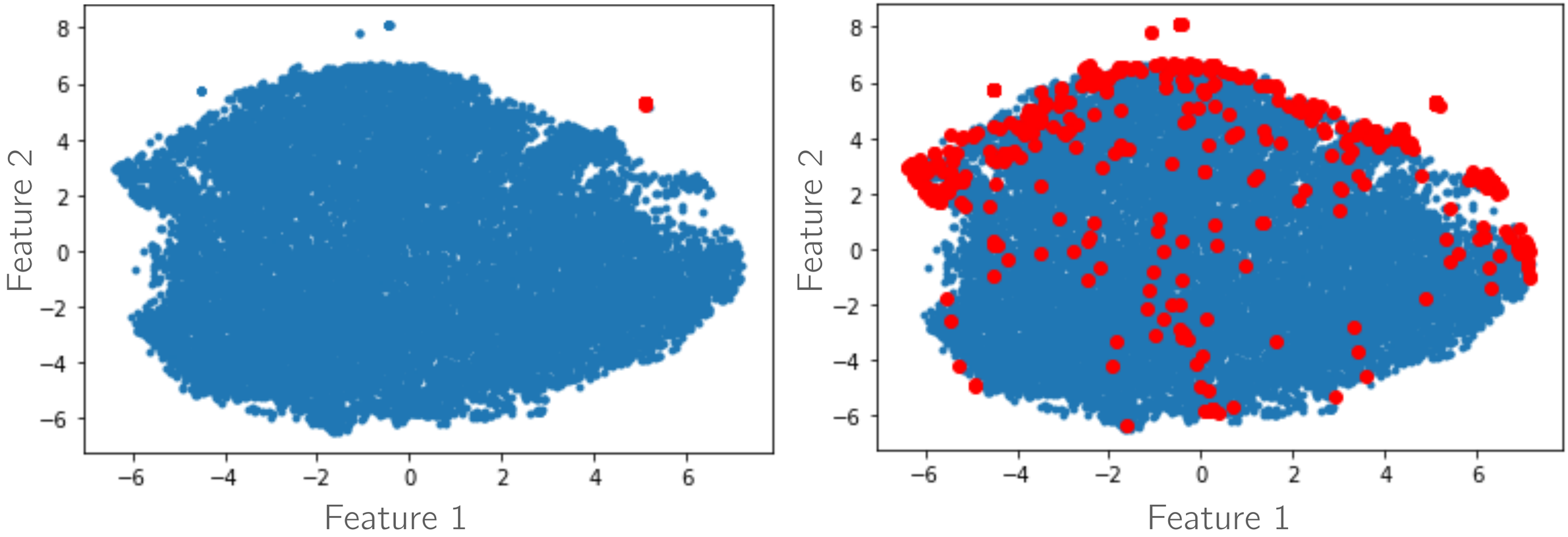



$l = 300$ length document vectors projected in 2D plane

# Project Outcomes and Discussion

## Unsupervised Model




$l = 300$ length document vectors projected in 2D plane

# Project Outcomes and Discussion

## Data Quality

- ASME corpus data is not great

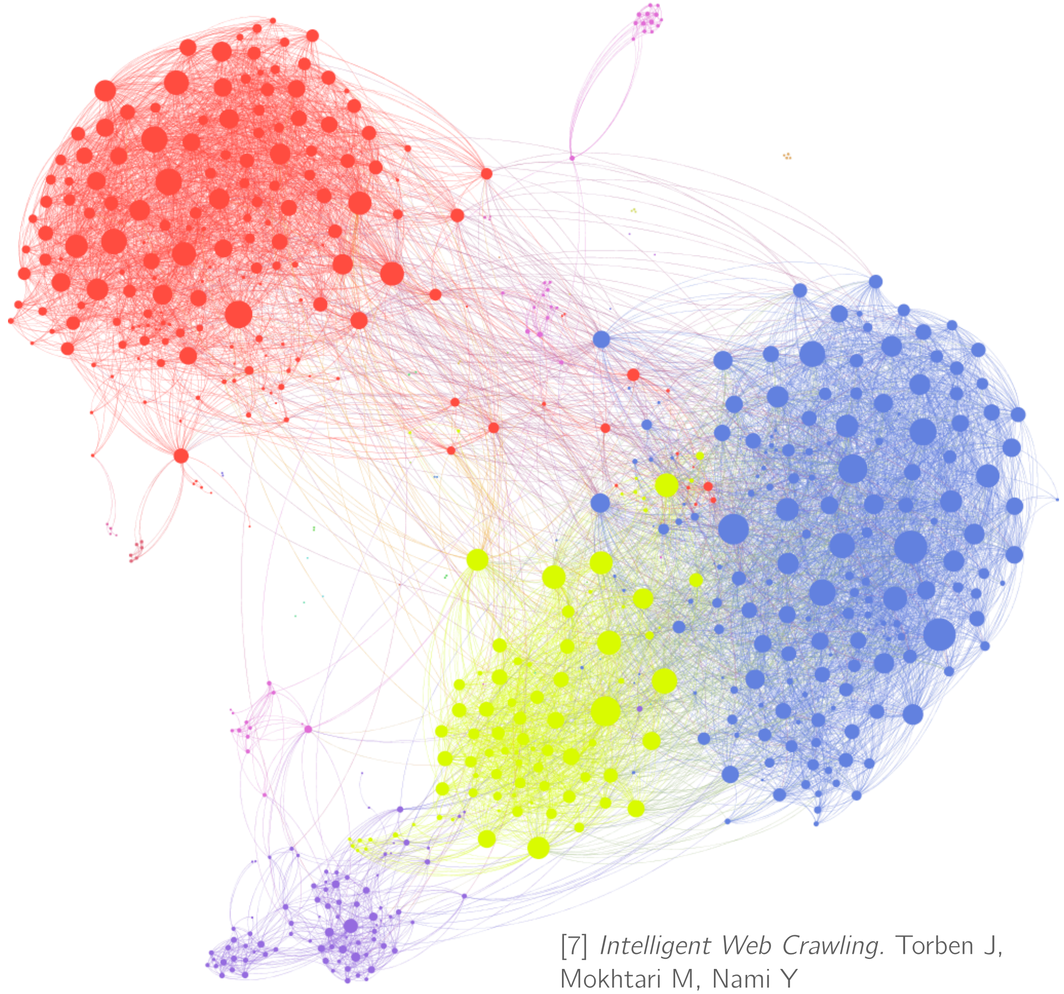
| Type | Actual word | Unrecognized word |
|---|---|---|
| Spelling mistakes | capturing | capaturing |
| Incorrectly read characters | copyright | capyright |
| Foreign words | | brennkammersystem |
| Maths symbols | $b \sin y$ | bsiny |
| Phrases captured without whitespace | burner loudspeaker setup | burnerloudspeakersetup |
| Named entities that weren't removed | | mohammadpour |
| Technical terms | | eulerian |

# Conclusion

- Two models: 1 pseudo-supervised, 1 unsupervised
- Pseudo-supervised model: highly computationally expensive, best left for pre-labelled and small sized documents
- Unsupervised model: Performance predictable, not consistent at detecting novelty
- Document representations not informative enough due to low quality data and choice of embeddings (GloVe, TF-IDF)
- TF-IDF is useful for detecting relevance, but not novelty

# Future Work

- **Redefining novelty:** considering graphical representations of documents for extracting information
- A novel research trend could be a branch that leads to another branch from a different cluster
- Can also examine novelty in graphical structures



[7] *Intelligent Web Crawling.* Torben J, Mokhtari M, Nami Y

# Future Work

- **Data enhancement:** use of TF-IDF as filtration step for noise removal, spell checking
- **Improved document representations:** use of state-of-the-art document representation techniques, including document encoder-decoder models
- **Computational considerations:** use of `tensorflow`'s `tf.Dataset` for better GPU acceleration
- **References:** using citations as extra features for training

Imperial College London

June 9 2021

# Thank you for your attention

Any Questions

# Reference List


[1] Digital Bibliography and Library project. *Publications per year.* Available from: dblp.org/statistics/publicationsperyear.html [Accessed 5th June 2021]

[2] FloydHub. *Introduction to Anomaly Detection.* Available from: blog.floydhub.com/introduction-to-anomaly-detection-in-python/ [Accessed 5th June 2021]

[3] Pennington J, Socher R, Manning CD. Global Vectors for Word Representation. Available from: nlp.stanford.edu/projects/glove/ [Accessed 5th June 2021]

[4] Sarkar D. Hands on Approach to Deep learning methods for text data. Available from: towardsdatascience.com/understanding-feature-engineering-part-4-deep-learning-methods-for-text-data-96c44370bbfa [Accessed 5th June 2021]

[5] Hui J. *Word Embedding and GloVe.* Available from: https://jonathan-hui.medium.com/nlp-word-embedding-glove-5e7f523999f6 [Accessed 5th June 2021]

[6] Qin Q, Hu W, Liu B. Text Classification with Novelty Detection. Arxiv [Preprint] 2020. Available from: https://arxiv.org/abs/2009.11119 [Accessed 5th June 2021]

[7] Torben J, Mokhtari R, Nami Y. *Intelligent Web Crawling* [Accessed 6th June 2021]