A Pipeline for Scalable Text Reuse Analysis

Milad Alshomary

Bauhaus Universität

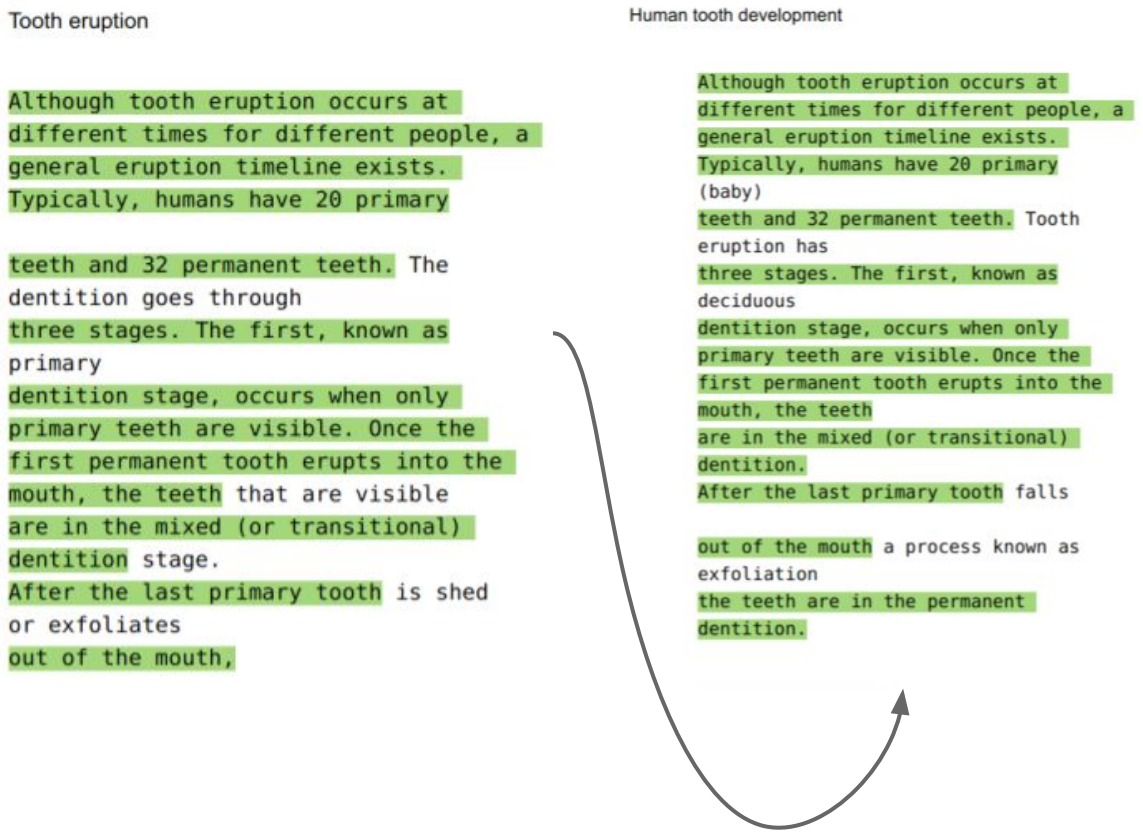
05.07.2018

Overview

- Motivation
- A Pipeline for Scalable Text Reuse Extraction
- Application on Wikipedia
- Application on Wikipedia and Common Crawl
- Conclusion

Text Reuse (TR)

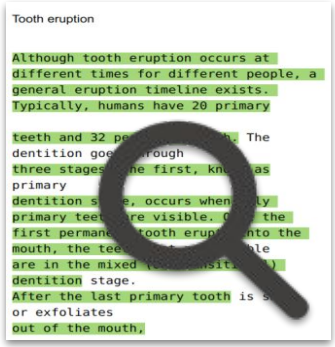
- Quoting
- Verbatim
- Paraphrasing
- Translation
- Summarization



TR Detection Applications

eTRAP
Electronic Text Reuse Acquisition Project

METER project (Measuring Text Reuse)

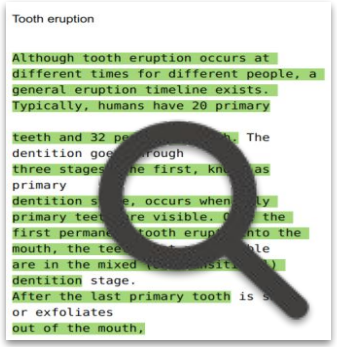


Plagiarism detection

TR Detection Applications

eTRAP
Electronic Text Reuse Acquisition Project

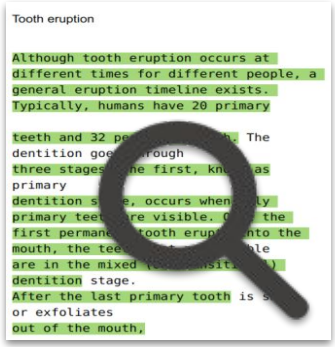
METER projet (Measuring Text Reuse)



Plagiarism detection

TR Detection Applications

METER projet (Measuring Text Reuse)



Plagiarism detection

Wikipedia vs The World

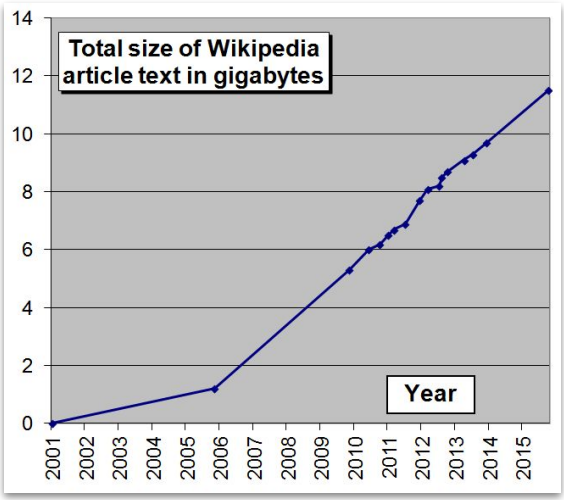
- Digital Encyclopedia
- Collaborative environment
- Giant public source of information
- Free to use

Wikipedia vs The World

- Digital Encyclopedia
- Collaborative environment
- Giant public source of information
- Free to use

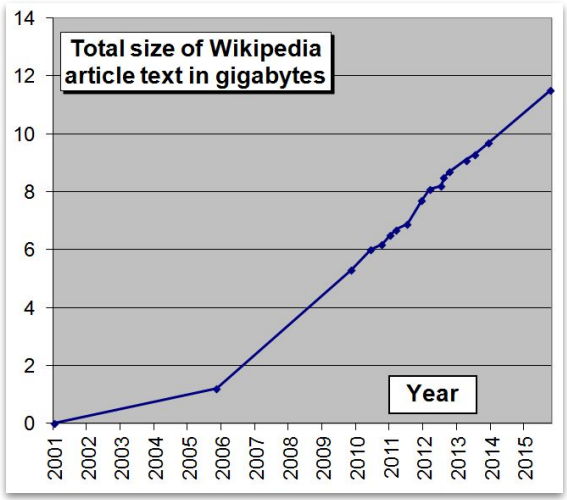
Wikipedia vs The World

- Digital Encyclopedia
- Collaborative environment
- Giant public source of information
- Free to use

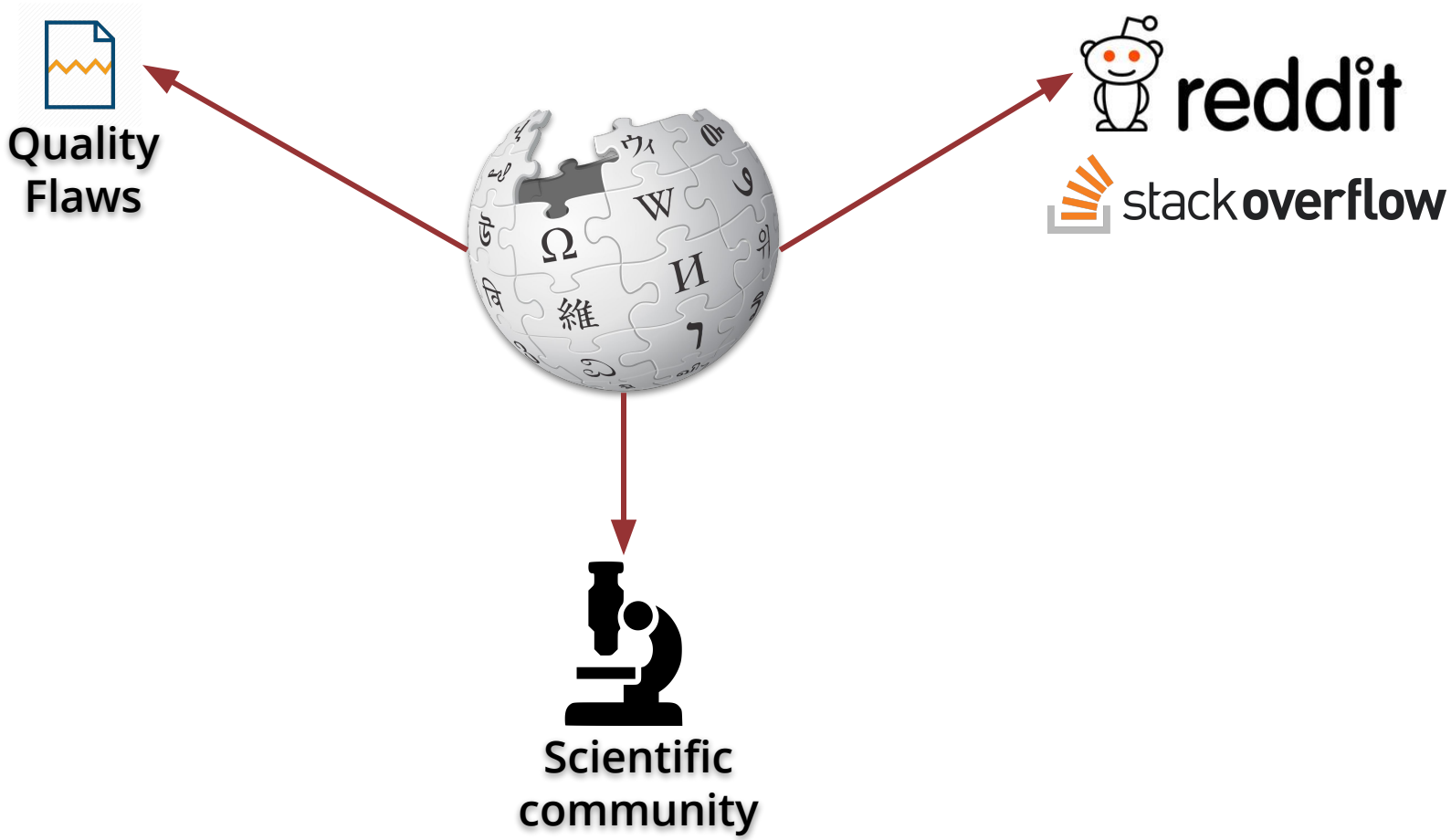


Wikipedia vs The World

- Digital Encyclopedia
- Collaborative environment
- Giant public source of information
- Free to use

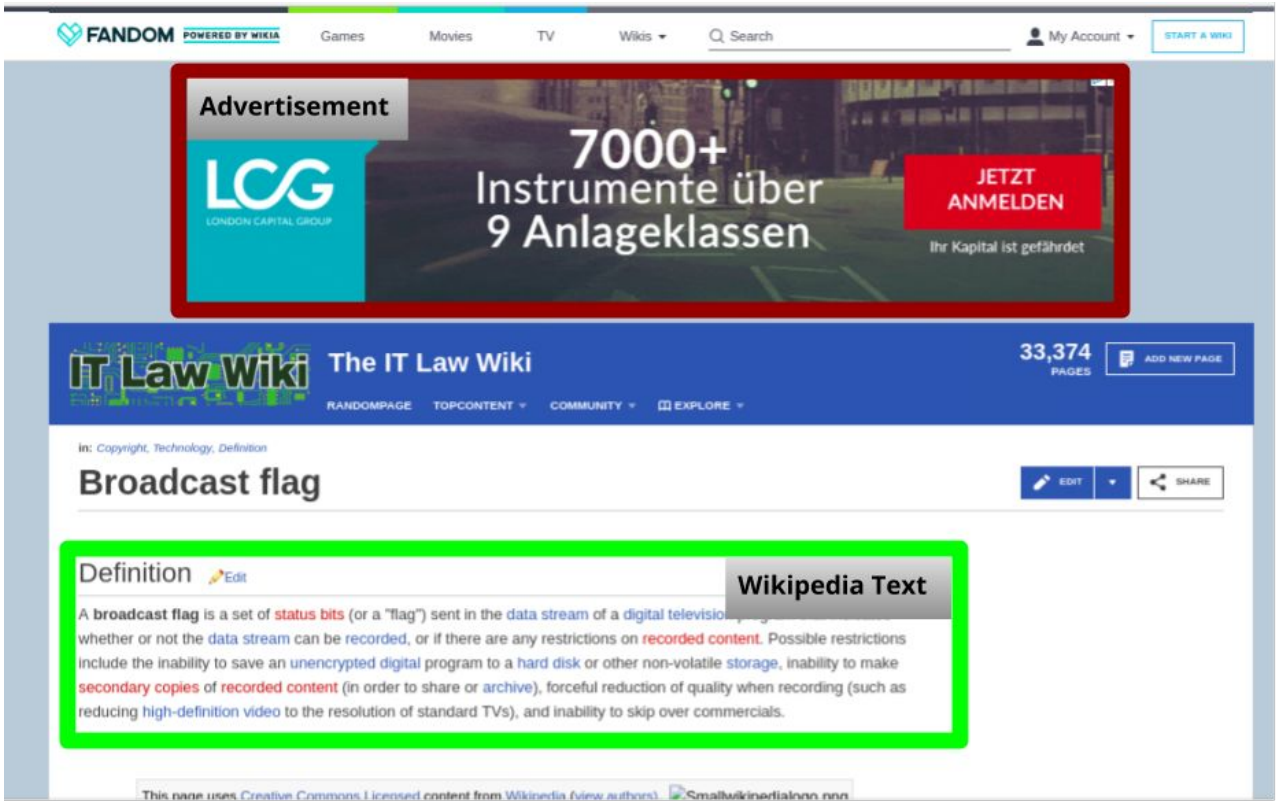


Wikipedia vs The World



Wikipedia vs The World

- Web pages = Wikipedia text + advertisements



Research Questions

➔ What kinds of text reuse occur within Wikipedia?

➔ How much of the web is a copy of Wikipedia content?

➔ How much revenue does this content generate?

Research Questions

➔ What kinds of text reuse occur within Wikipedia?

➔ How much of the web is a copy of Wikipedia content?

➔ How much revenue does this content generate?

Research Questions

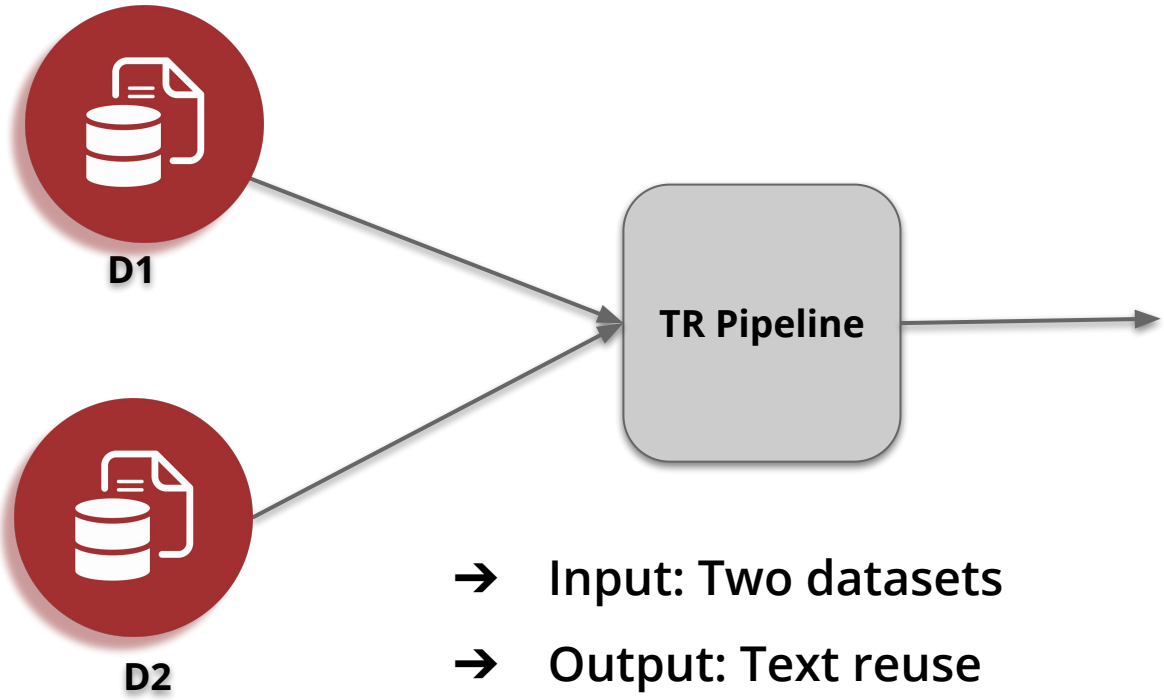
➔ What kinds of text reuse occur within Wikipedia?

➔ How much of the web is a copy of Wikipedia content?

➔ How much revenue does this content generate?

A Pipeline for Scalable Text Reuse Extraction

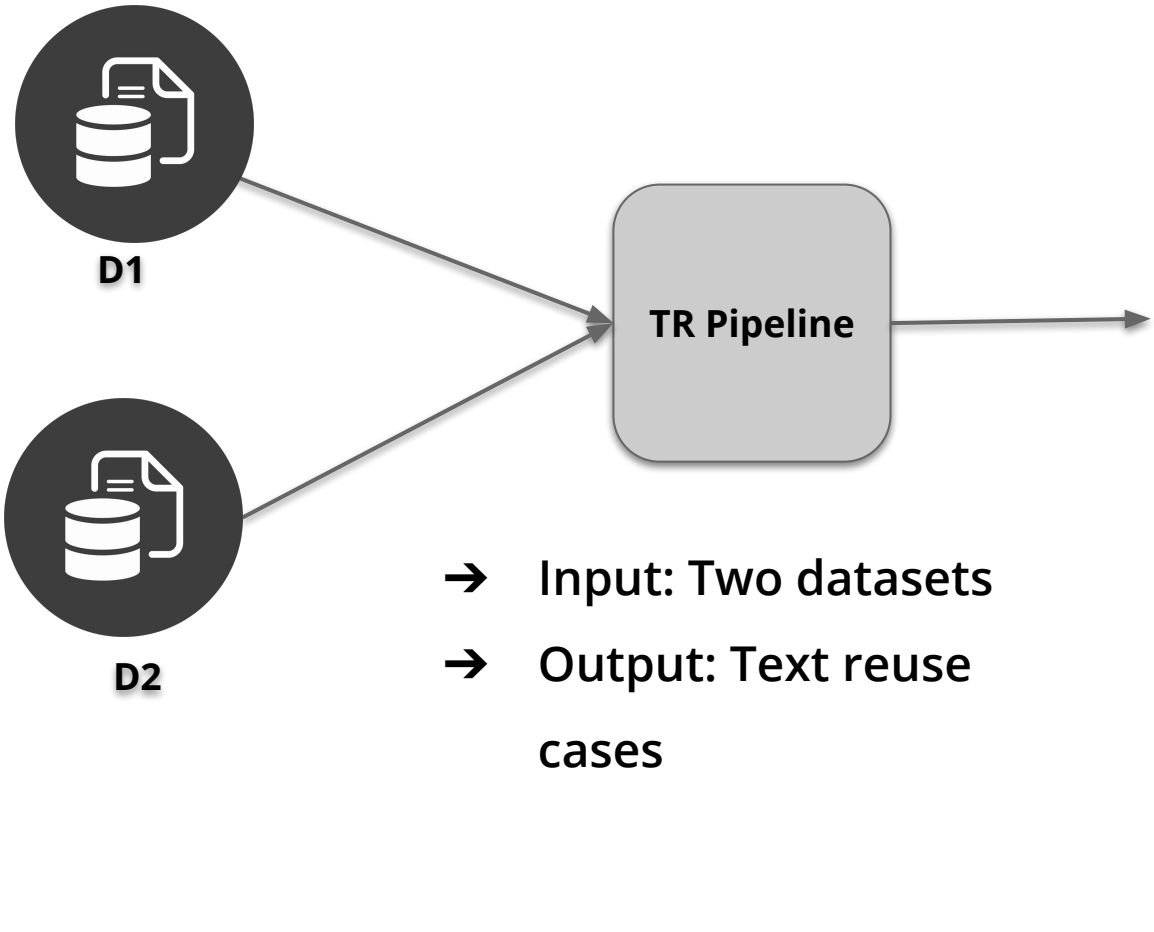
Text Reuse Pipeline



➔ Input: Two datasets

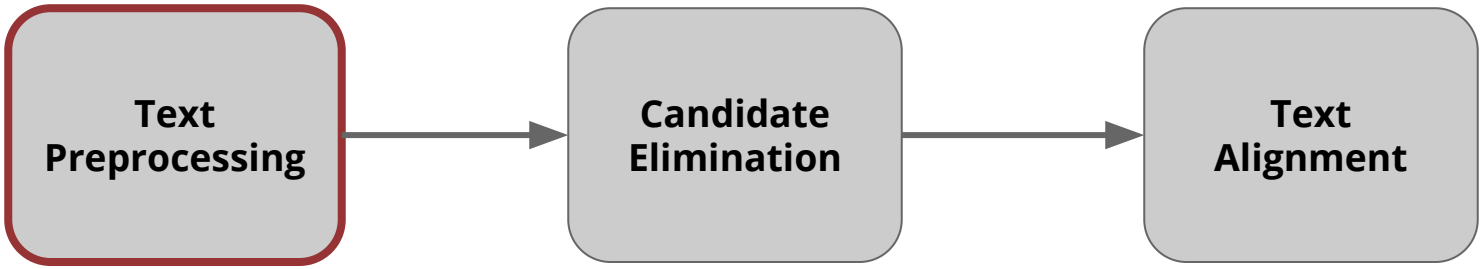
➔ Output: Text reuse cases

Text Reuse Pipeline

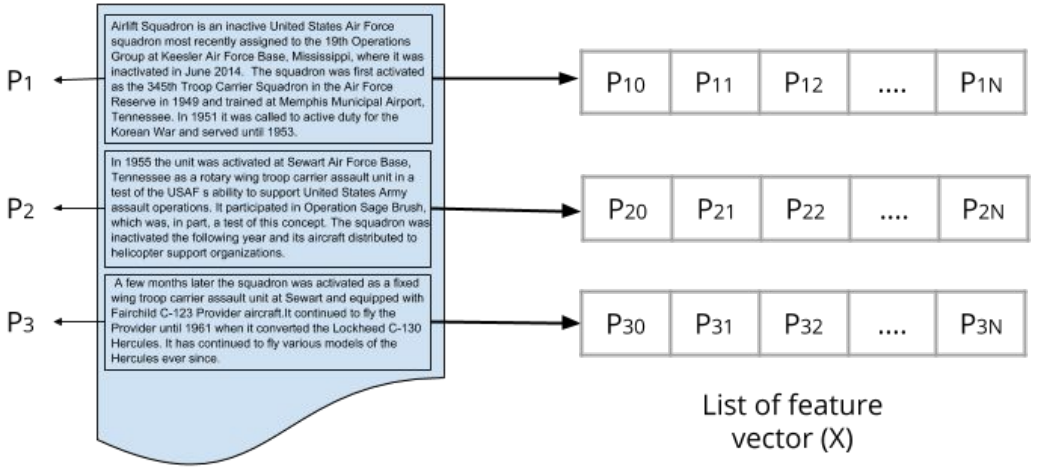


- ➔ Input: Two datasets
- ➔ Output: Text reuse cases

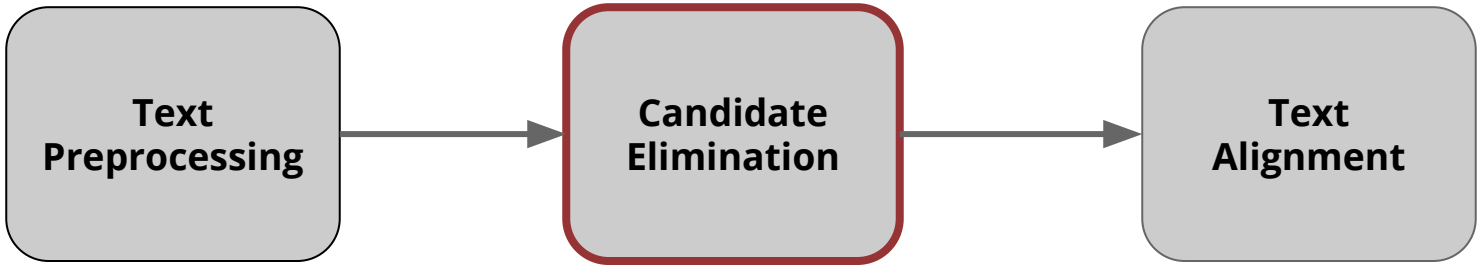
Text Reuse Pipeline



- ➔ Content extraction
- ➔ Chunking
- ➔ Feature extraction



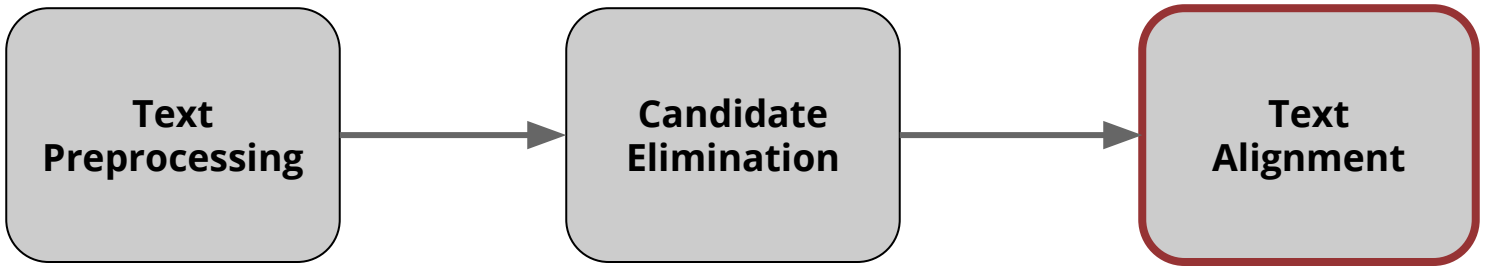
Text Reuse Pipeline



- ➔ Content extraction
- ➔ Chunking
- ➔ Feature extraction

- ➔ Pairwise scan
- ➔ Text Reuse heuristics

Text Reuse Pipeline

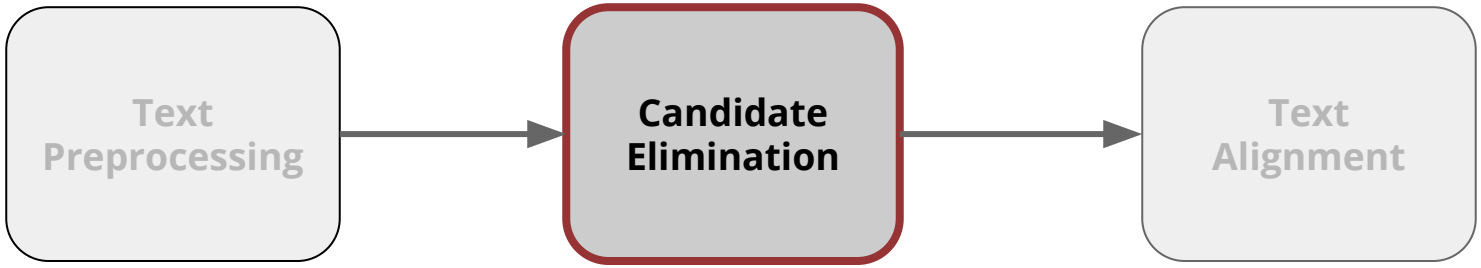


- ➔ Content extraction
- ➔ Chunking
- ➔ Feature extraction

- ➔ Pairwise scan
- ➔ Text Reuse heuristics

- ➔ Detailed scan of text reuse
- ➔ Picapica framework

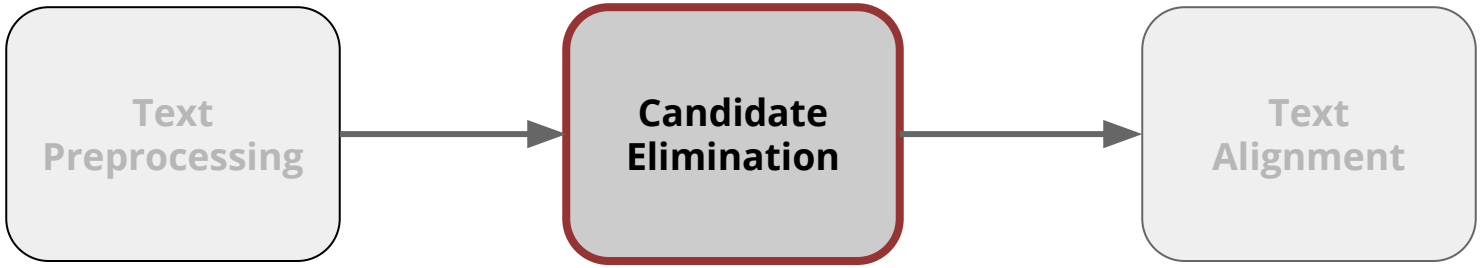
Candidate Elimination



Keys for scaling-up:

- ➔ Cluster computing
- ➔ Heuristics based candidate elimination algorithms

Candidate Elimination



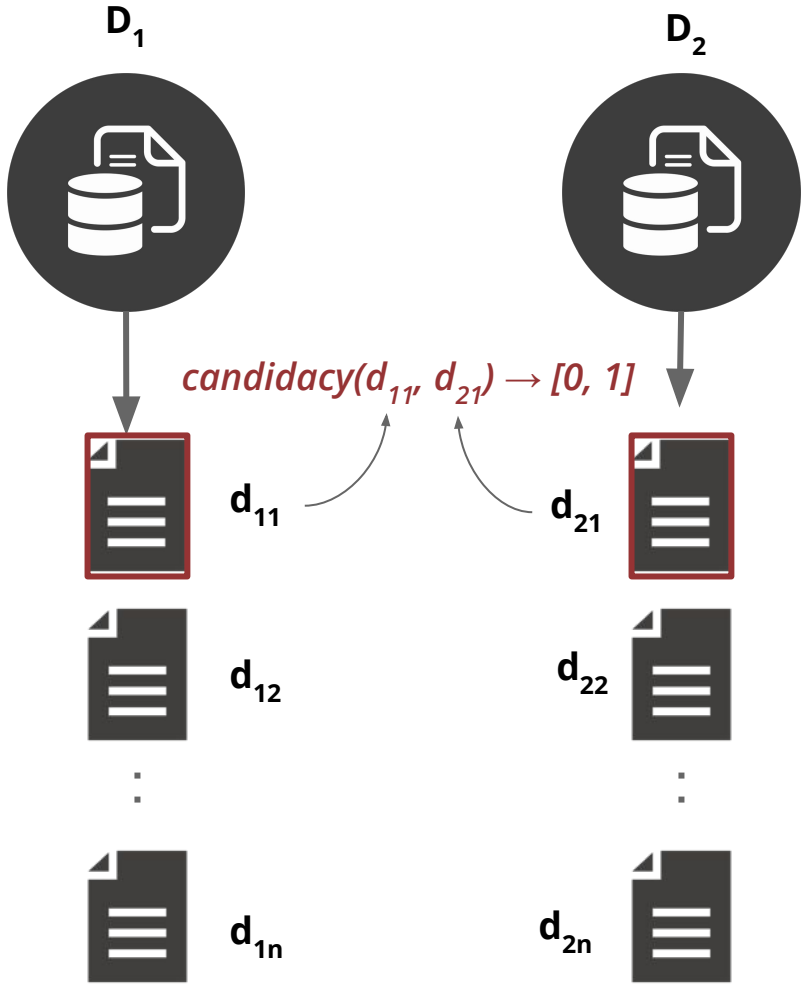
Keys for scaling-up:

- ➔ **Cluster computing**
- ➔ **Heuristics based candidate elimination algorithms**

Candidate Elimination

For a *candidacy* function we proposed the following methods:

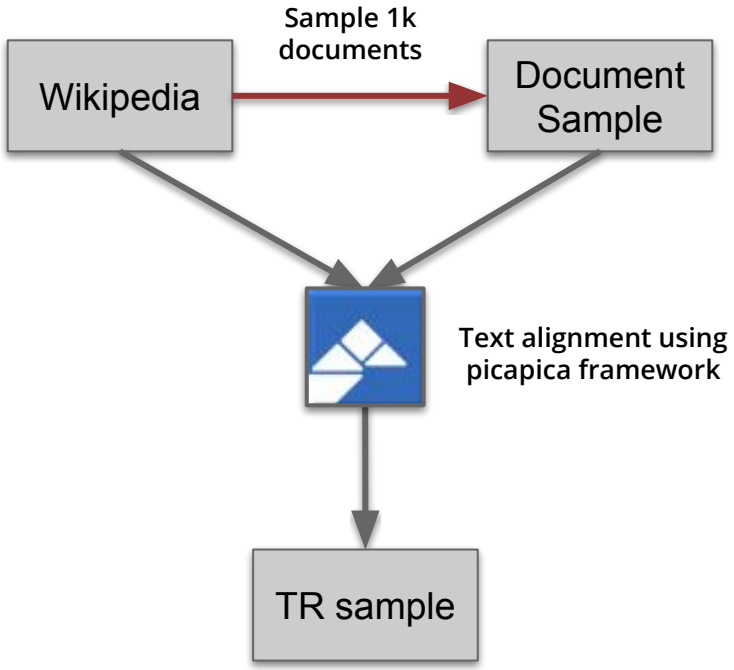
- Cosine similarity of TF-IDF (semantic)
- Paragraph embedding (semantic)
- Stopwords N-grams (structure)
- Weighted average of Stopwords Ngrams and Paragraph embedding (semantic + structure)



Candidate Elimination

Generate TR Sample from Wikipedia:

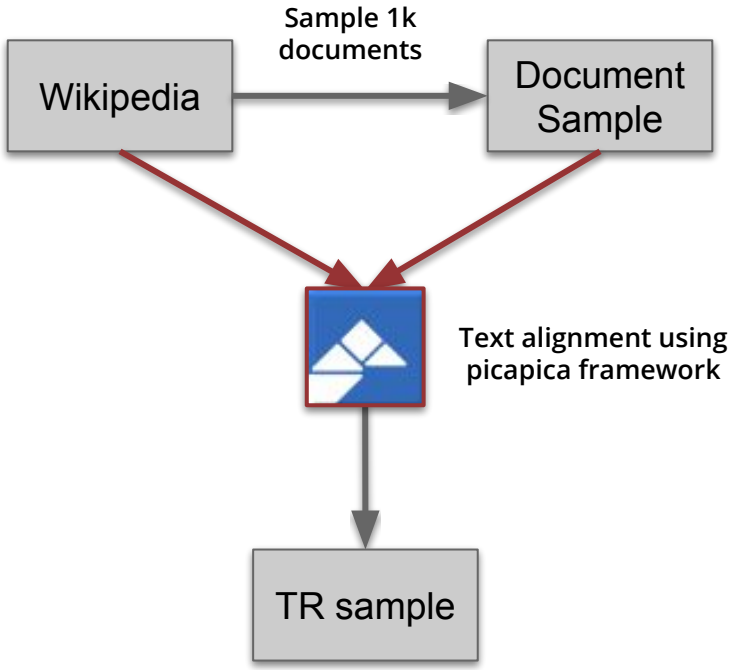
- Sample 1k documents from Wikipedia
- Using Picapica framework to find TR cases



Candidate Elimination

Generate TR Sample from Wikipedia:

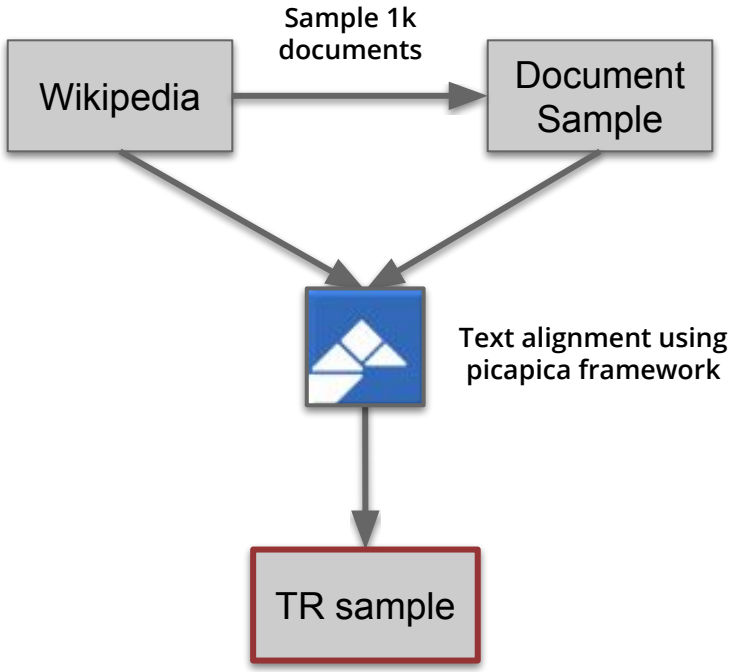
- **Sample 1k documents from Wikipedia**
- **Using Picapica framework to find TR cases**

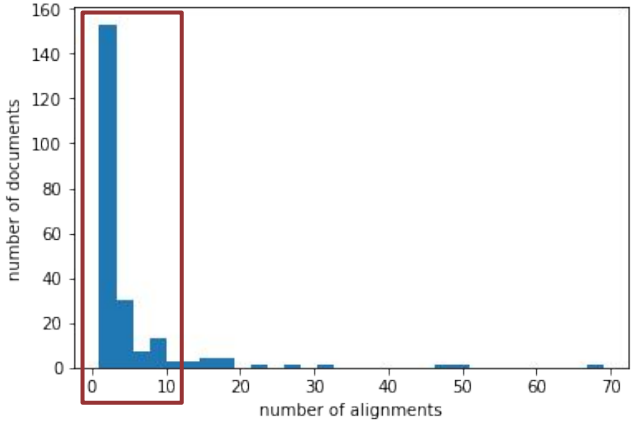


Candidate Elimination

Generate TR Sample from Wikipedia:

- **Sample 1k documents from Wikipedia**
- **Using Picapica framework to find TR cases**



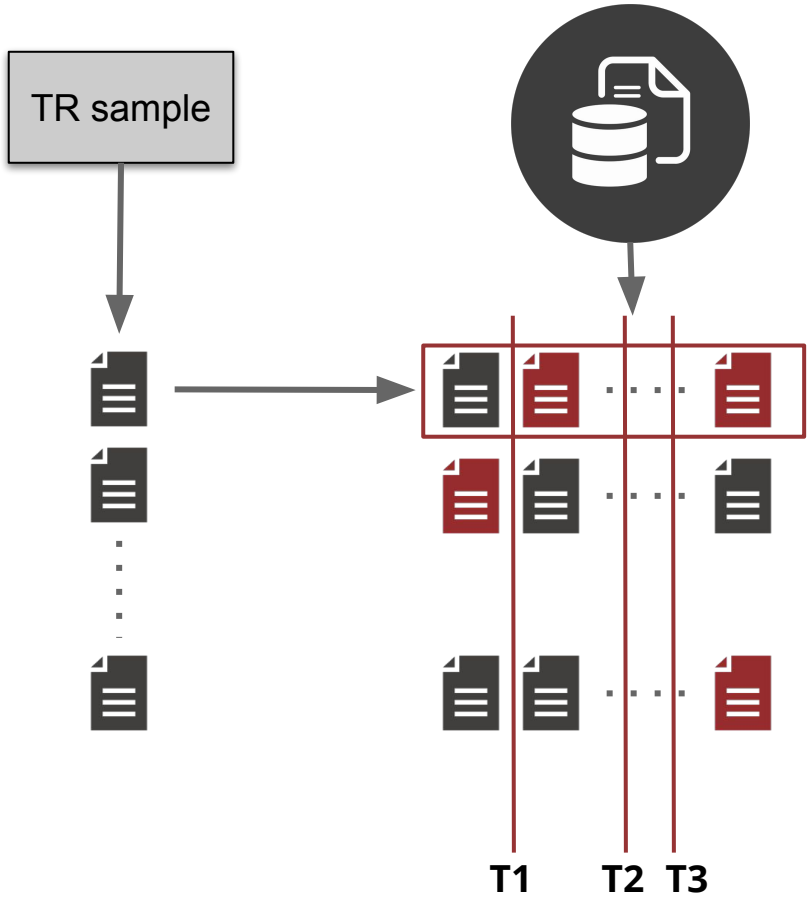


- **232 documents**
- **~ 90% have < 10 alignements (TR case)**

Candidate Elimination

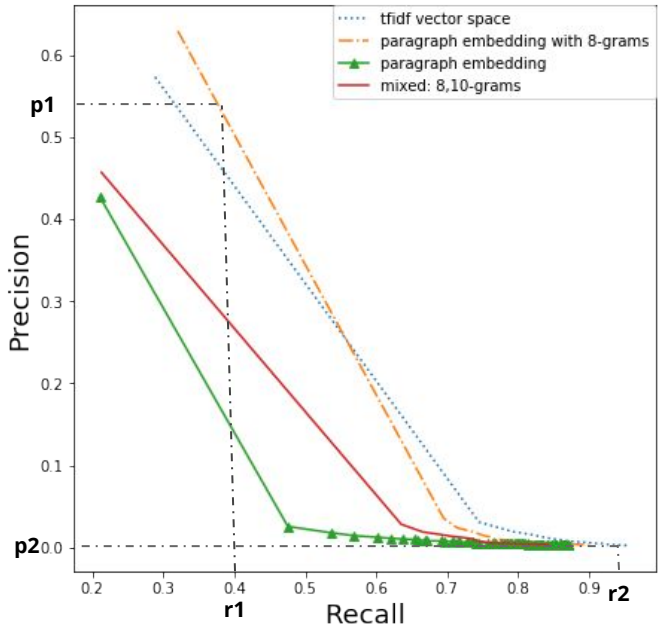
Evaluation of "*candidacy*" function:

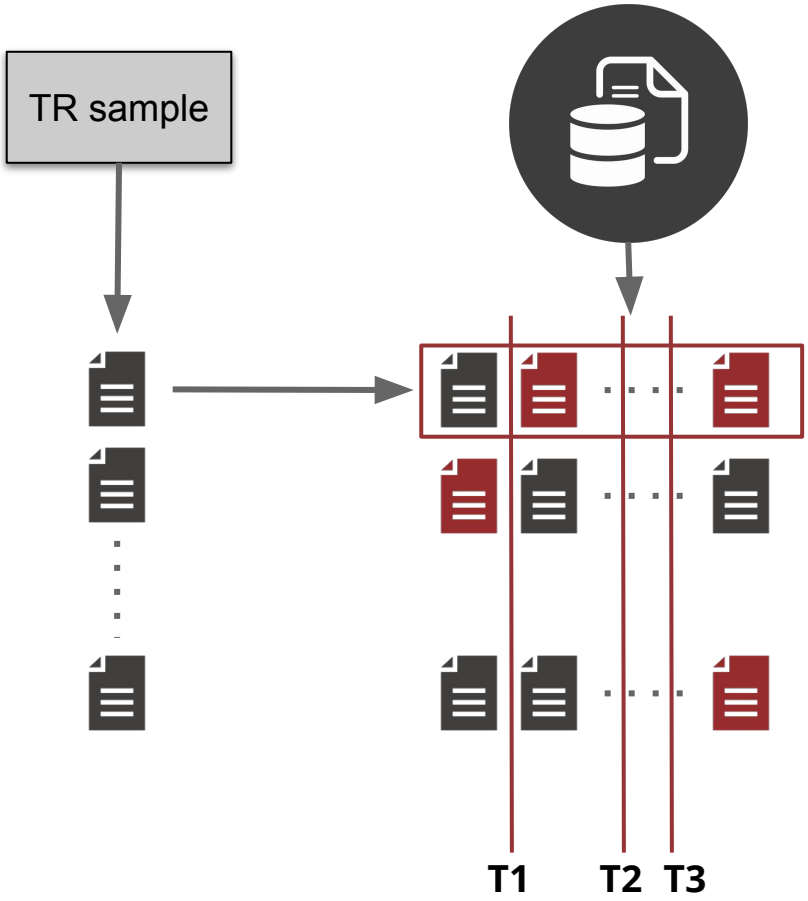
- For each document in TR sample:
  - Sort all Wikipedia articles according to the proposed "*candidacy*" .
  - Precision/Recall on Thresholds of [1, 101,..,100k]
  - A True Positive (TP) is a pair of documents that have TR.



Candidate Elimination

Evaluation of "*candidacy*" function:

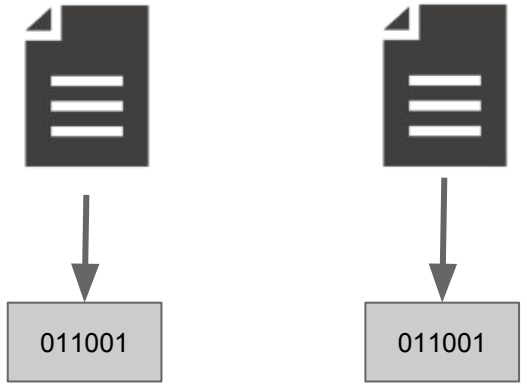




Candidate Elimination

Semantic hashing function:

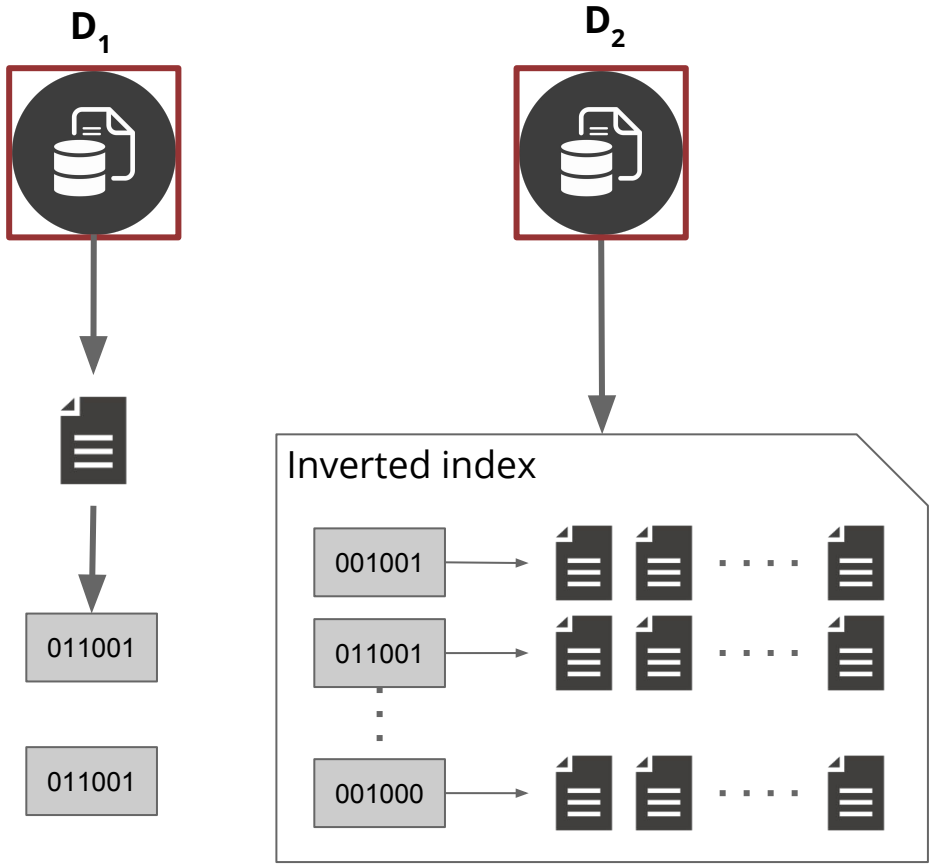
- **Hashes documents into binary hashes.**
- **Similar documents get similar or exact binary hash.**



Candidate Elimination

Semantic hashing function:

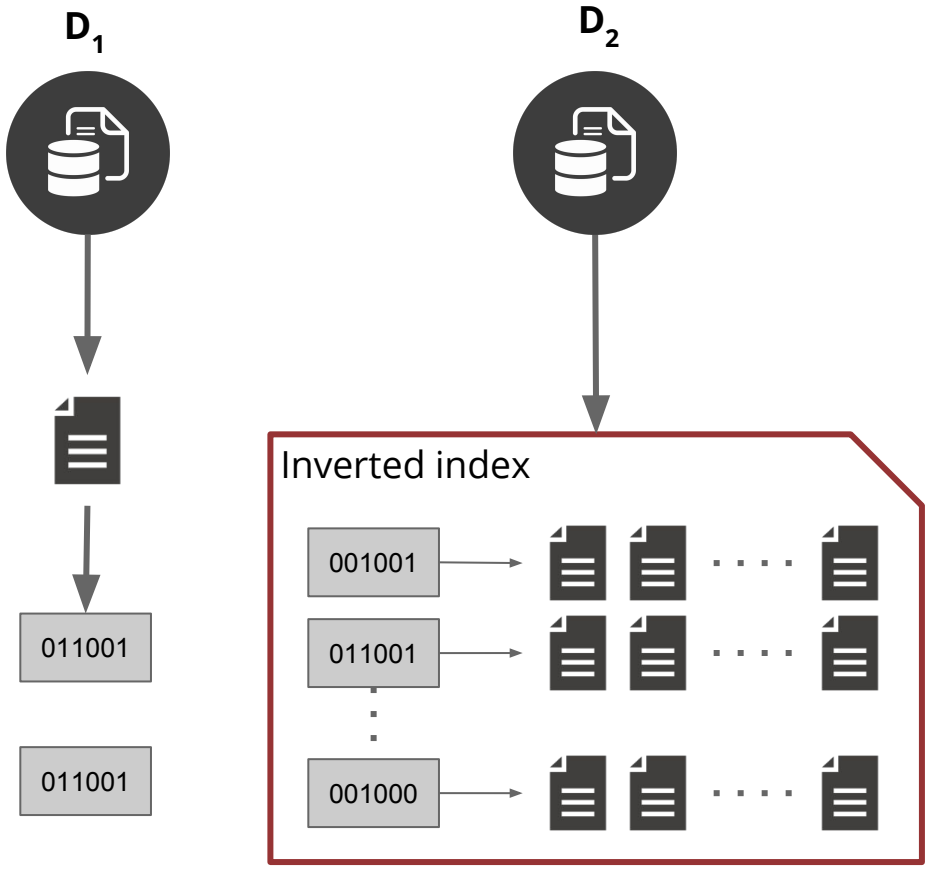
- Hashing all documents.
- Inverted index.
- Hash document's chunks.
- Apply *candidacy* function only on documents that intersect in one hash at least.



Candidate Elimination

Semantic hashing function:

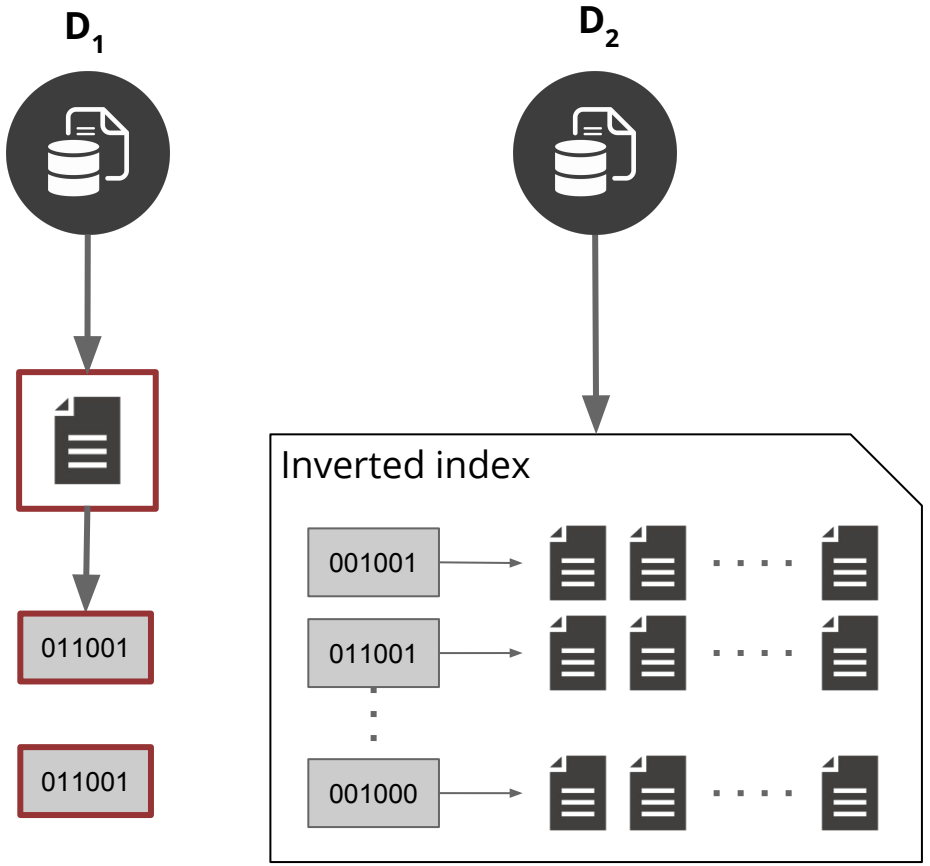
- Hashing all documents.
- Inverted index.
- Hash document's chunks.
- Apply *candidacy* function only on documents that intersect in one hash at least.



Candidate Elimination

Semantic hashing function:

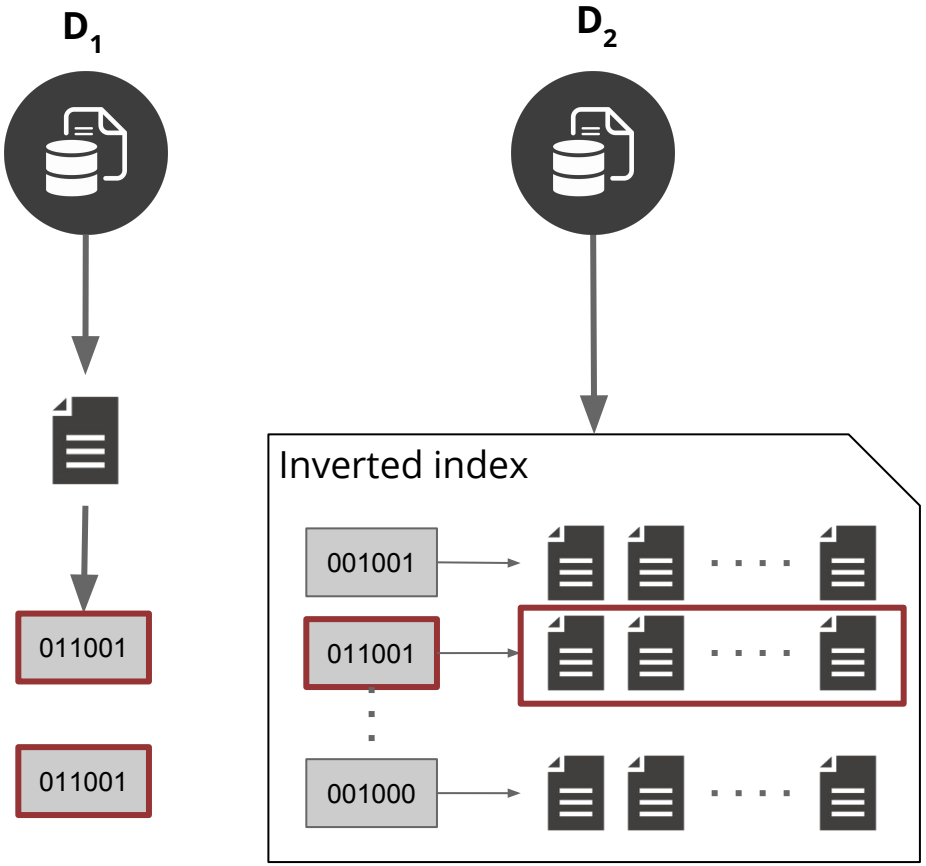
- **Hashing all documents.**
- **Inverted index.**
- **Hash document's chunks.**
- **Apply *candidacy* function only on documents that intersect in one hash at least.**



Candidate Elimination

Semantic hashing function:

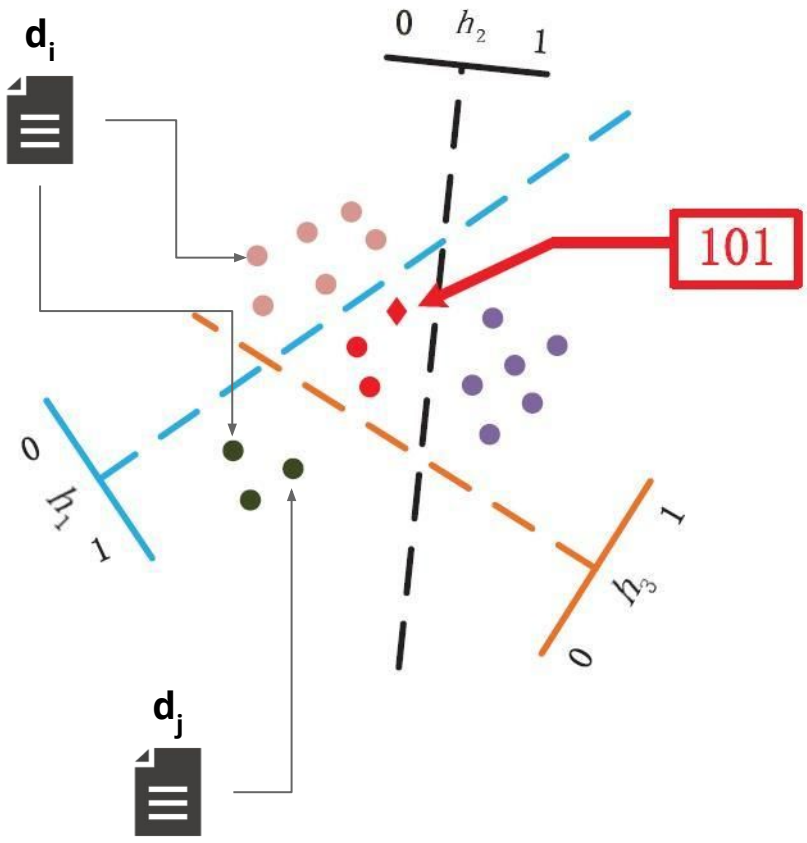
- Hashing all documents.
- Inverted index.
- Hash document's chunks.
- Apply *candidacy* function only on documents that intersect in one hash at least.



Candidate Elimination

Proposed semantic hashing methods:

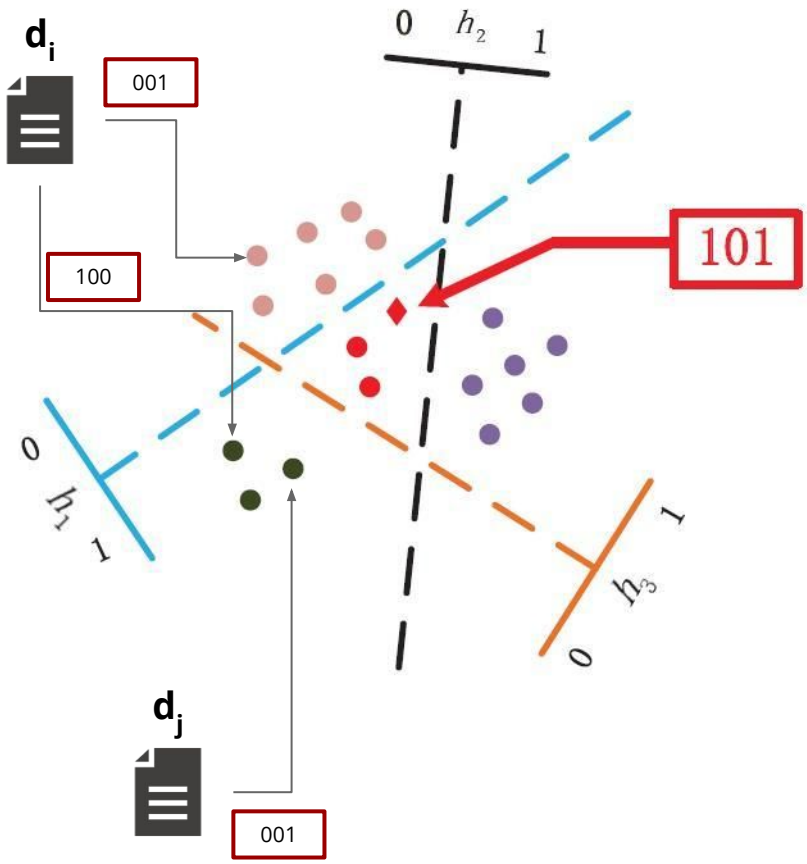
- Random Projection (data independent)
- Variational Deep Semantic Hashing (data dependent)



Candidate Elimination

Proposed semantic hashing methods:

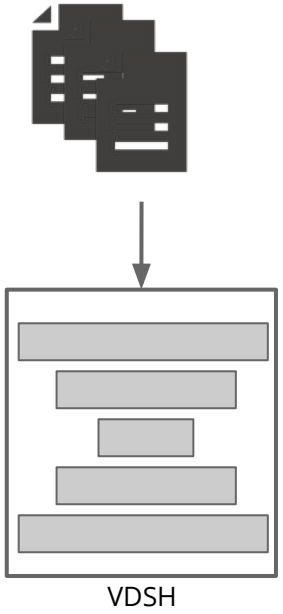
- Random Projection (data independent)
- Variational Deep Semantic Hashing (data dependent)



Candidate Elimination

Proposed semantic hashing methods:

- **Random Projection (data independent)**
- **Variational Deep Semantic Hashing (data dependent)**

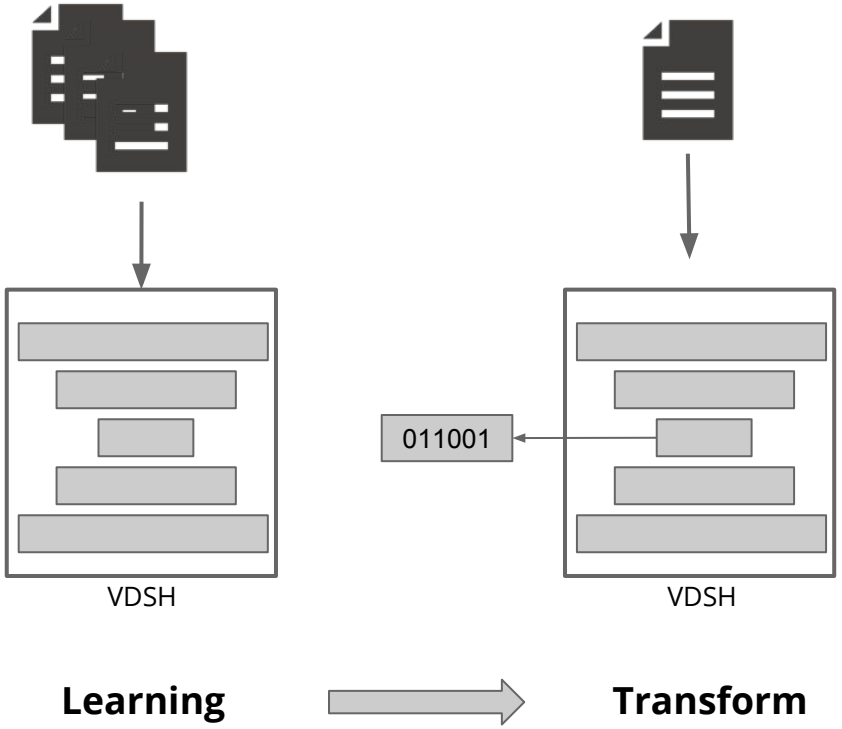


Learning

Candidate Elimination

Proposed semantic hashing methods:

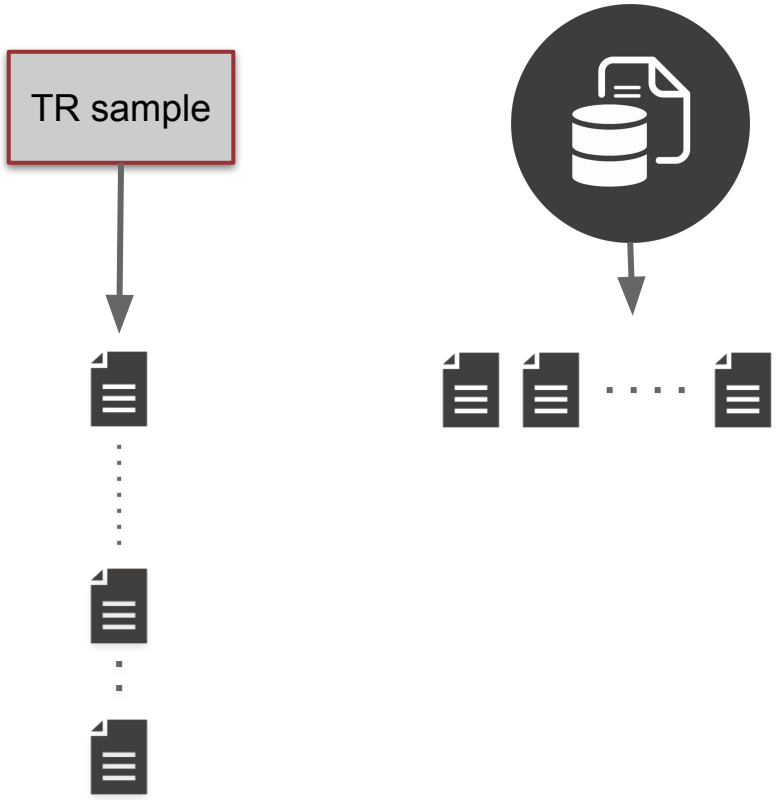
- **Random Projection (data independent)**
- **Variational Deep Semantic Hashing (data dependent)**



Candidate Elimination

Hashing methods evaluation:

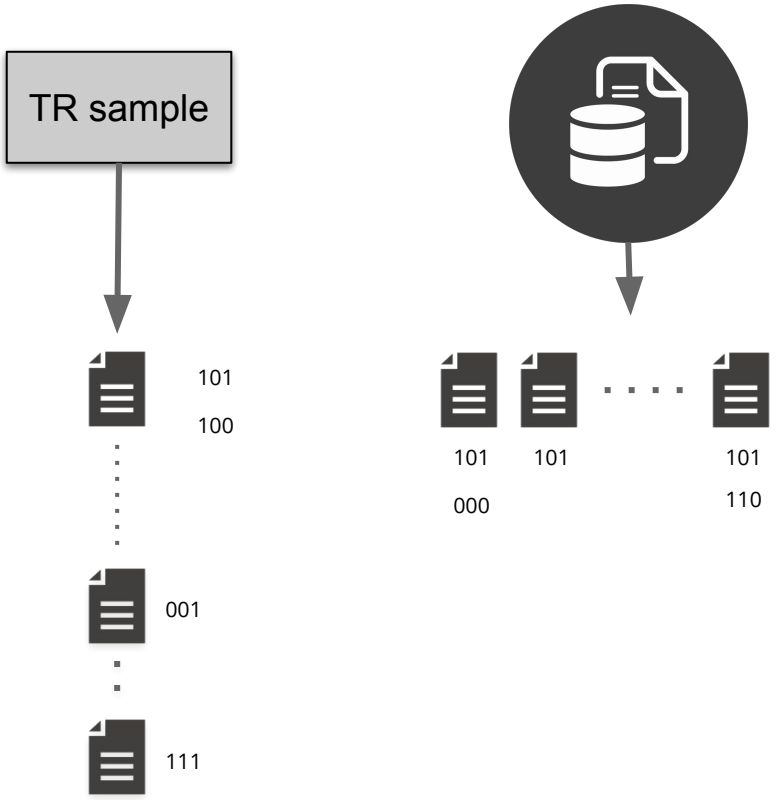
- Using same TR sample for evaluation.
- Hashing all documents using the proposed hashing function.
- Compute precision and recall.



Candidate Elimination

Hashing methods evaluation:

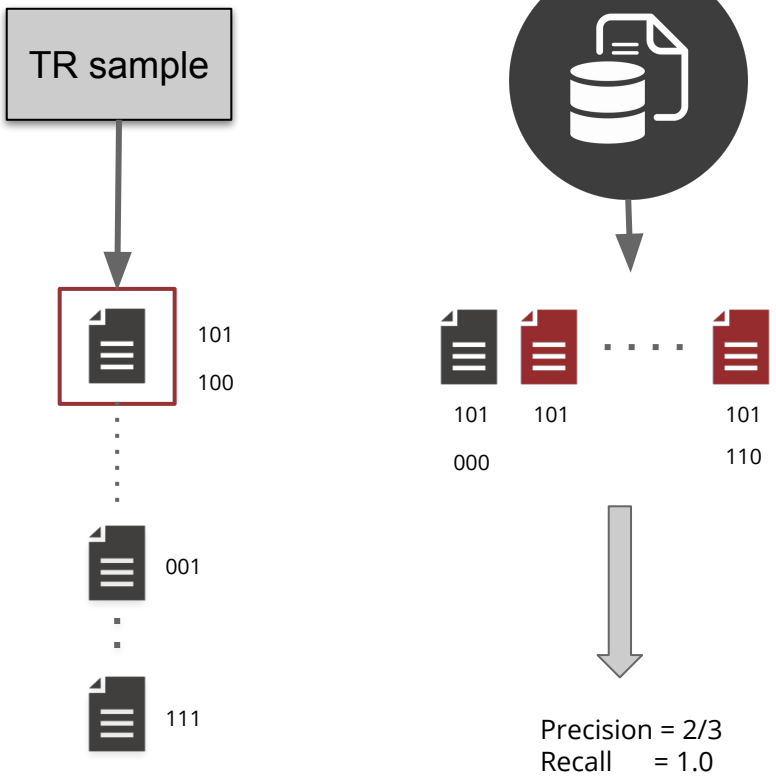
- **Using same TR sample for evaluation.**
- **Hashing all documents using the proposed hashing function.**
- **Compute precision and recall.**



Candidate Elimination

Hashing methods evaluation:

- **Using same TR sample for evaluation.**
- **Hashing all documents using the proposed hashing function.**
- **Compute precision and recall.**



Candidate Elimination

Hashing methods evaluation

- Using same TR sample for evaluation.
- Hashing all documents using the proposed hashing function.
- Compute precision and recall.

Random projection	bits	precision	recall
….	8	3.1×10^{-4}	0.8741
….	16	9.9×10^{-4}	0.324

VDSH	bits	precision	recall
….	8	2.8×10^{-4}	0.88
….	16	4.5×10^{-3}	0.73

Candidate Elimination

Hashing methods evaluation

- Using same TR sample for evaluation.
- Hashing all documents using the proposed hashing function.
- Compute precision and recall.

VDSH	bits	precision	recall
	8	2.8×10^{-4}	0.88
	16	4.5×10^{-3}	0.73

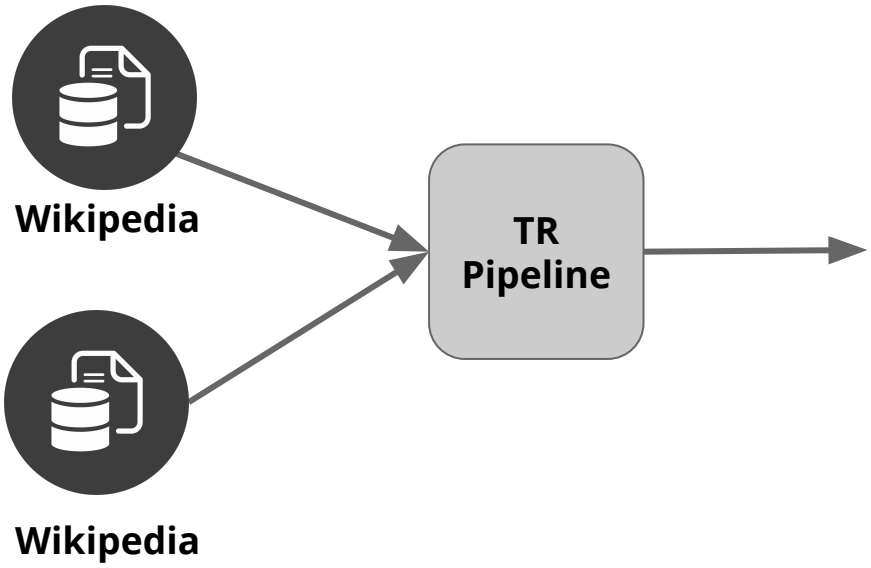
- Retains 73% of the recall
- By experiment:
  - Reduces the computations needed by 3 order of magnitude

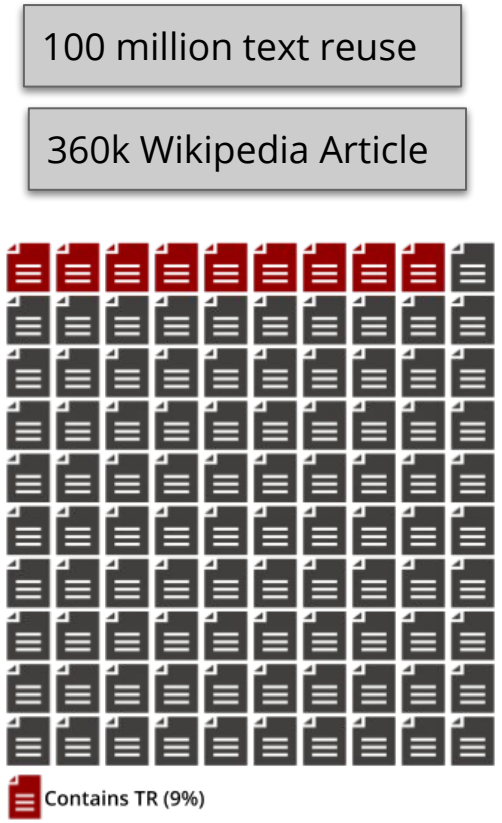
Application on Wikipedia

Text Reuse In Wikipedia

- ➔ What kinds of text reuse occur within Wikipedia?
- ➔ How much of the web is a copy of Wikipedia content?
- ➔ How much revenue does this content generate?

Text Reuse In Wikipedia





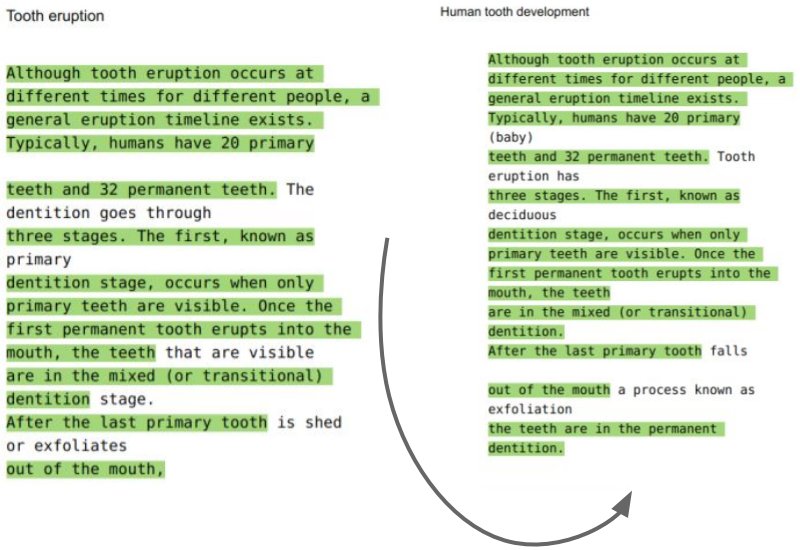
Text Reuse In Wikipedia

What kinds of text reuse occur in Wikipedia?

- **Reasons behind text reuse:**

  (1) Two texts describe the same topic.

  (2) Two texts describe two different topics, that share similar characteristics

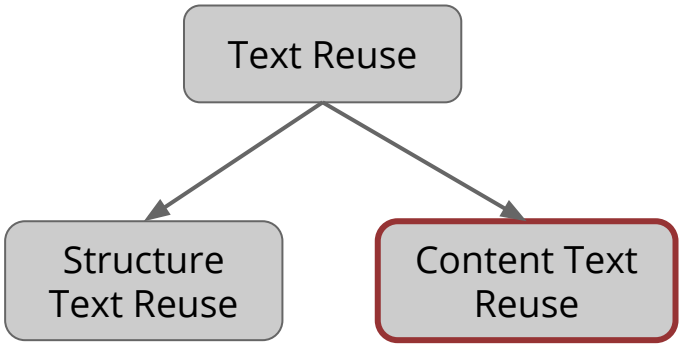


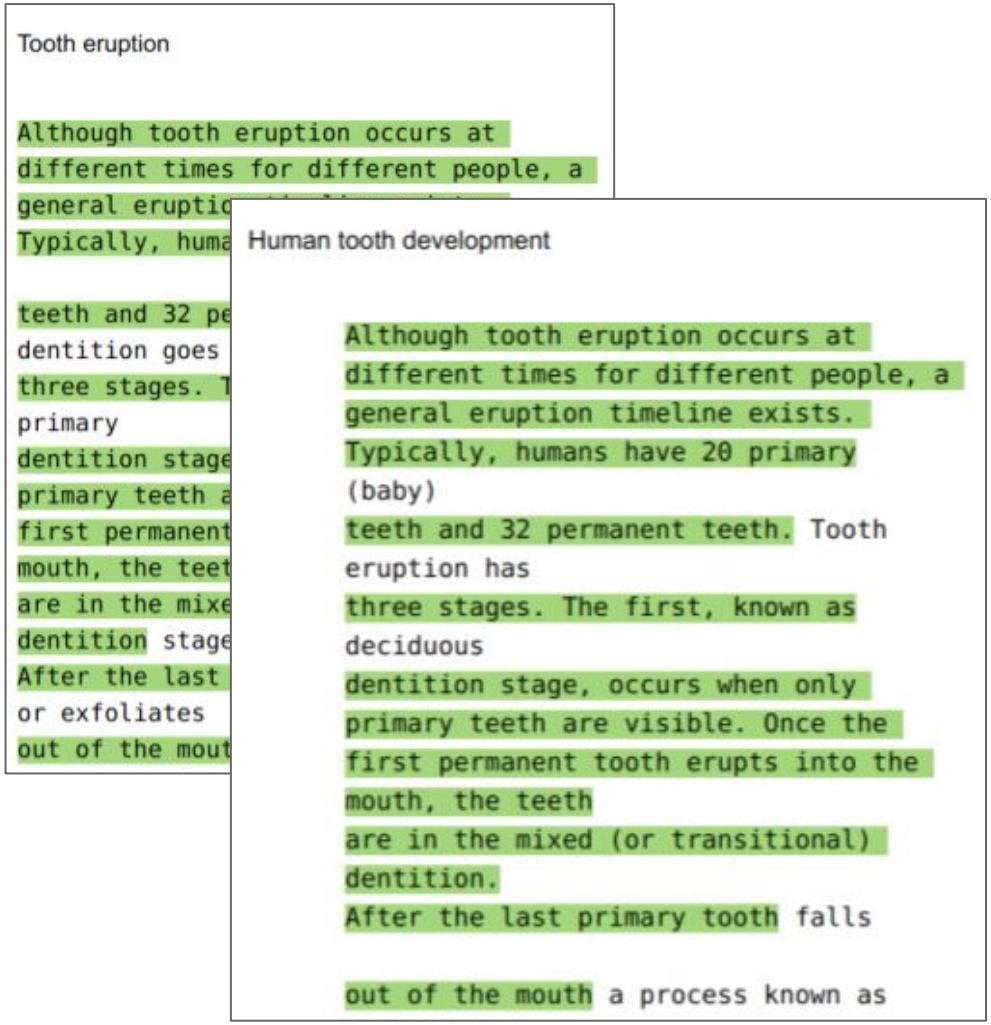
Text Reuse In Wikipedia

What kinds of text reuse occur in Wikipedia?

- **Reasons behind text reuse:**

  (1) Two texts describe the same topic.





Text Reuse In Wikipedia

What

Wikip

- **R**

Tooth eruption

Although tooth eruption occurs at different times for different people, a general eruption timeline exists. Typically, humans have 20 primary

teeth and 32 permanent teeth. The dentition goes through three stages. The first, known as primary dentition stage, occurs when only primary teeth are visible. Once the first permanent tooth erupts into the mouth, the teeth that are visible are in the mixed (or transitional) dentition stage. After the last primary tooth is shed or exfoliates out of the mouth,

Str
Tex

Human tooth development

Although tooth eruption occurs at different times for different people, a general eruption timeline exists. Typically, humans have 20 primary (baby) teeth and 32 permanent teeth. Tooth eruption has three stages. The first, known as deciduous dentition stage, occurs when only primary teeth are visible. Once the first permanent tooth erupts into the mouth, the teeth are in the mixed (or transitional) dentition. After the last primary tooth falls

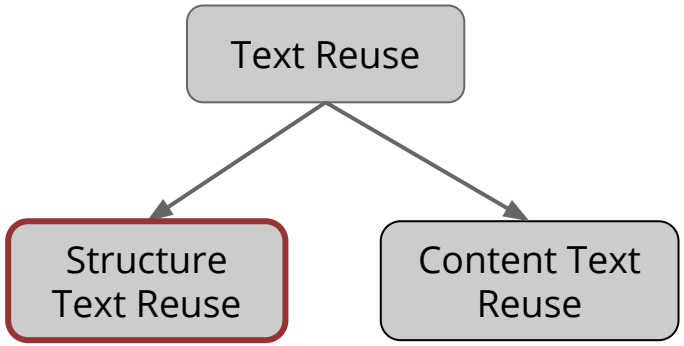
out of the mouth a process known as

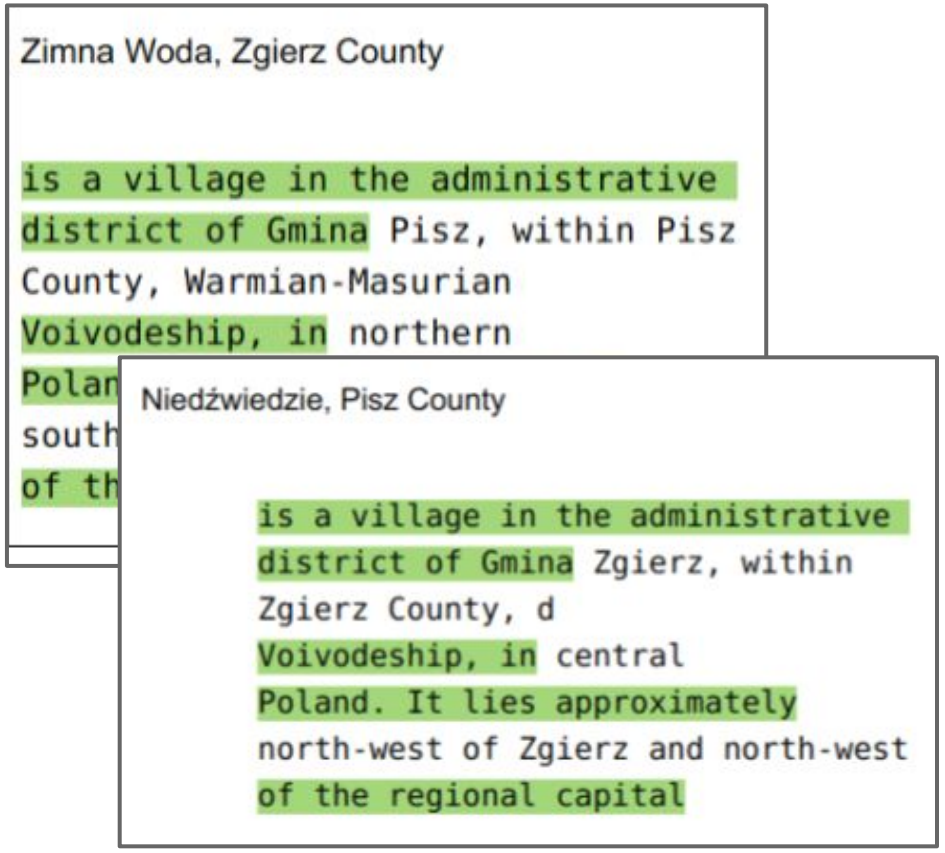
Text Reuse In Wikipedia

What kinds of text reuse occur in Wikipedia?

- **Reasons behind text reuse:**

  (2) Two texts describe two different topics, that share similar characteristics



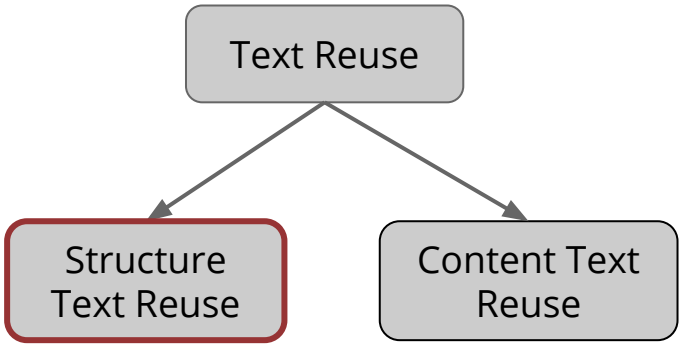


Text Reuse In Wikipedia

What kinds of text reuse occur in Wikipedia?

- **Reasons behind text reuse:**

  (2) Two texts describe two different topics, that share simila characteristics



Słowików, Opole Voivodeship

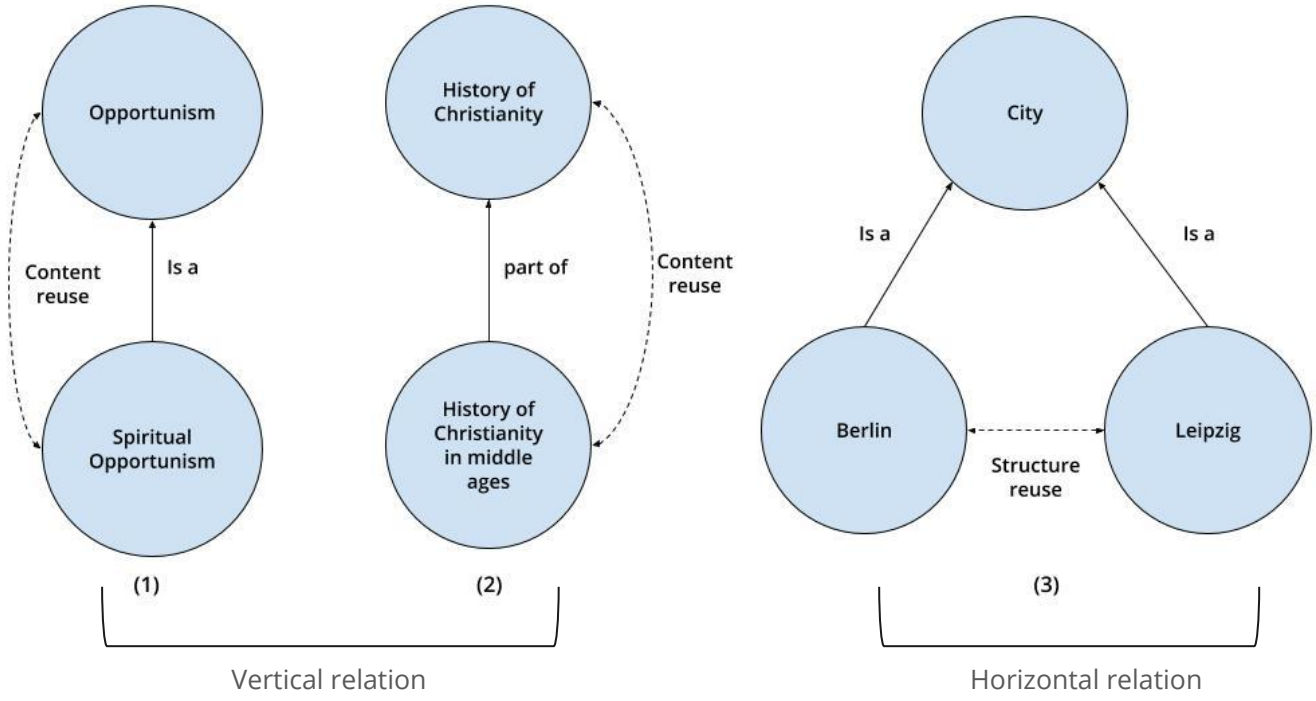
is a village in the administrative district of Gmina Rudniki, within Olesno County, Opole Voivodeship, in south-western Poland. It lies approximately east of Rudniki, north-east of Olesno, and north-east of the regional capital Opole. The village has a population of

Jaworzno, Opole Voivodeship

is a village in the administrative district of Gmina Rudniki, within Olesno County, Opole Voivodeship, in south-western Poland. It lies approximately north-east of Rudniki, north-east of Olesno, and north-east of the regional capital Opole. The village has a population of

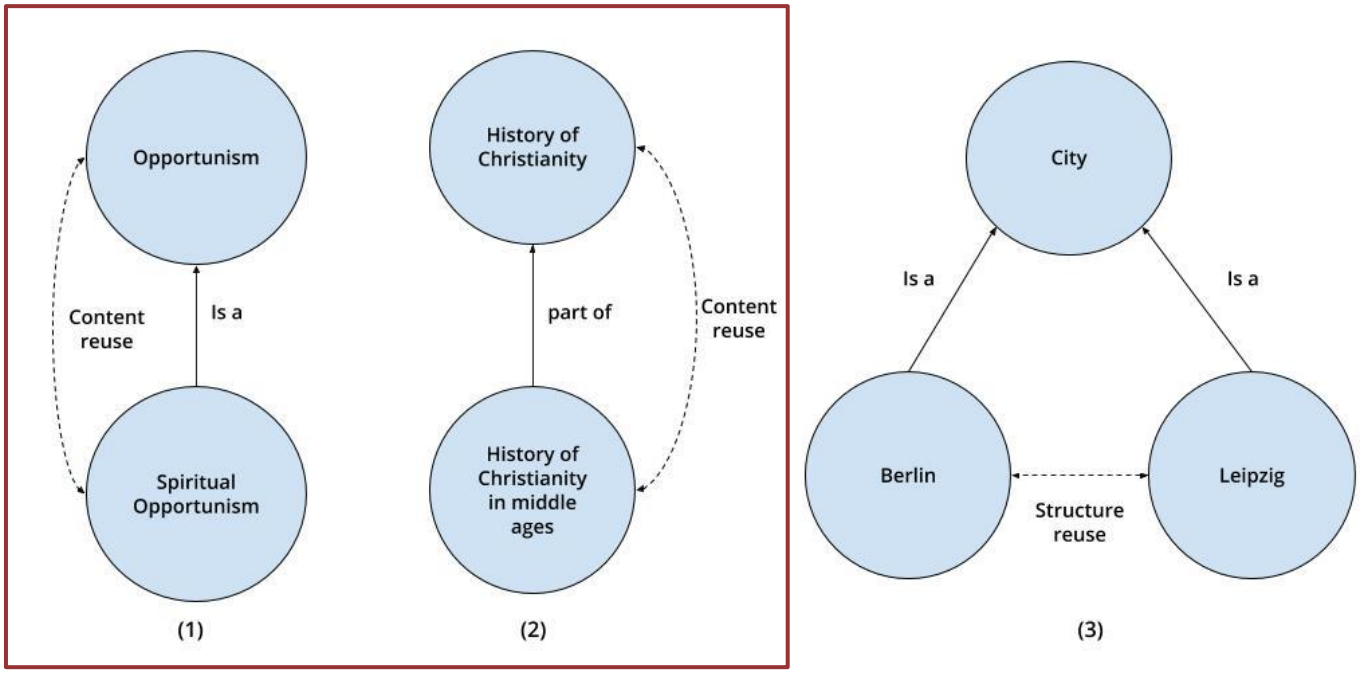
Text Reuse In Wikipedia

- **Vertical alignment → Content TR**
- **Horizontal alignment → Structure TR**



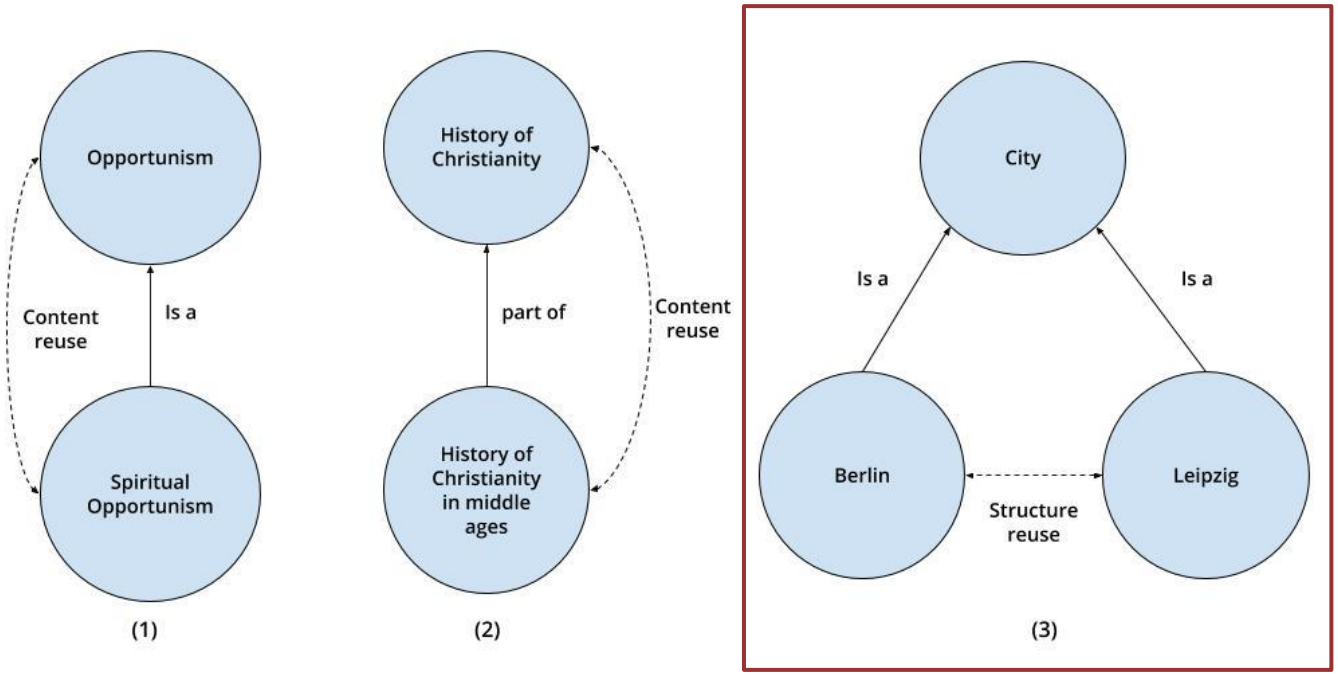
Text Reuse In Wikipedia

- Vertical alignment → Content TR
- Horizontal alignment → Structure TR



Text Reuse In Wikipedia

- **Vertical alignment → Content TR**
- **Horizontal alignment → Structure TR**

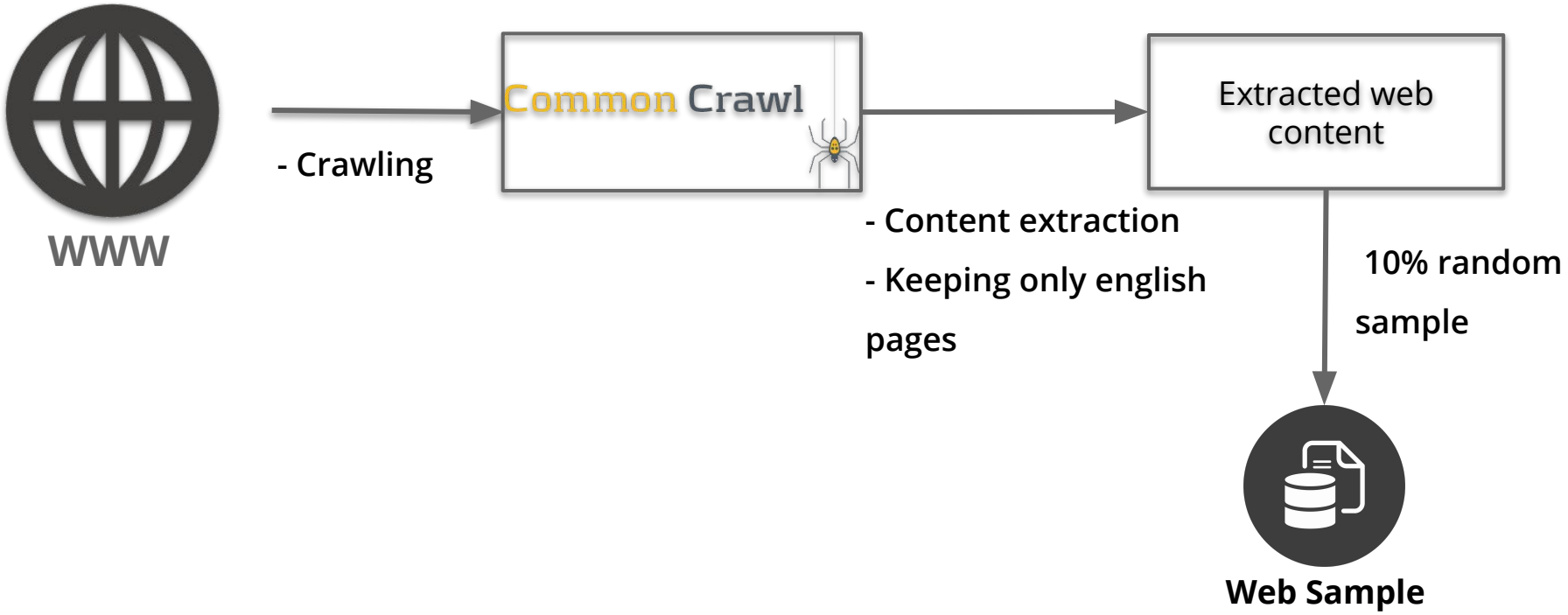


Application on Wikipedia and Common Crawl

Wikipedia vs The Web

- ➔ What kinds of text reuse occur within Wikipedia?
- ➔ How much of the web is a copy of Wikipedia content?
- ➔ How much revenue does this content generate?

Wikipedia vs The Web



Wikipedia vs The Web

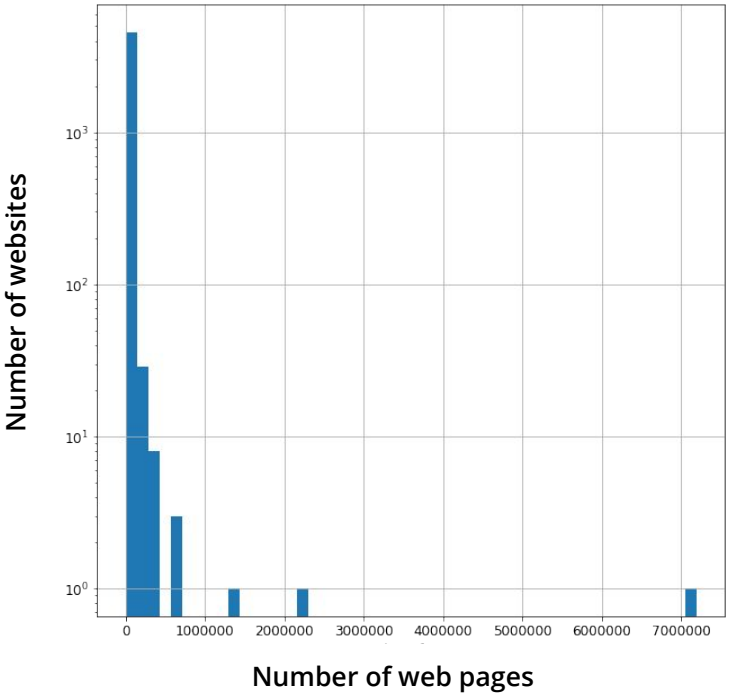
- Crawling

Extracted web content

- Content extraction
- Keeping only english pages

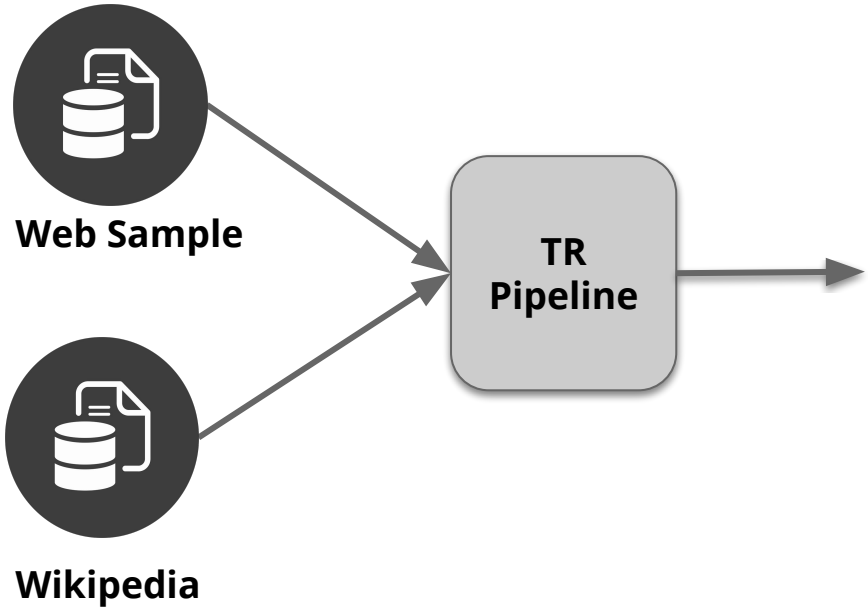
10% random sample

Web Sample

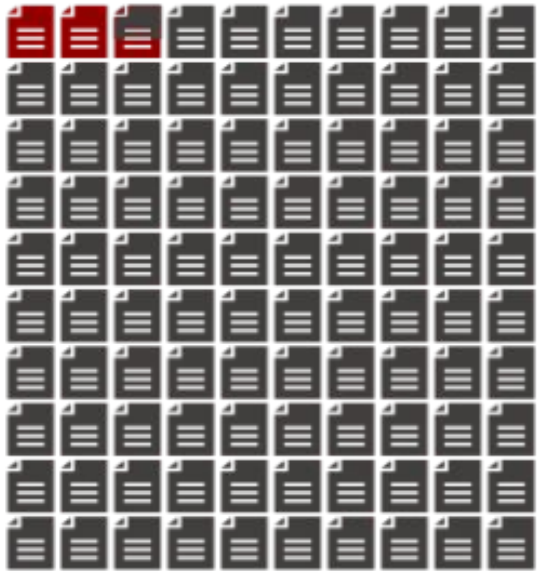


- 59 million web pages.
- 1.4 million websites.
- 70% of these websites contains less than 10 web pages

Wikipedia vs The Web



- 1.6 million text reuse cases.
- 15k pages reuse Wikipedia text.
- 4.8k websites reuse Wikipedia text.



Page with TR from Wikipedia (2.5%)

Bauhaus-Universität Weimar

Wikipedia vs The Web

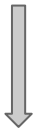
Monthly revenue estimation:

- **Rough estimate of Ads revenue**
- **Based on CPM (Cost Per Millie)**
- **Sampled 100 webpages and manually checked the existence of Advertisements.**

Wikipedia vs The Web

Revenue estimation:

- **Per website (all websites)**
- **Per website (highly reusing)**
- **Per Wikipedia web page**

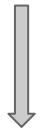


website	Monthly revenue	Percentage of reuse	Monthly Wikipedia value
pdxretro.com	$195	0.012	$2.5
seqrchquarry.com	$8,850	0.096	$850
asiatees.com	$36,000	0.017	$613
....	.....	.....	....
Total			$1.2 million

Wikipedia vs The Web

Revenue estimation:

- **Per website (all websites)**
- **Per website (highly reusing)**
- **Per Wikipedia web page**

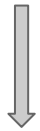


website	Monthly revenue	Percentage of reuse	Monthly Wikipedia value
pdxretro.com	$195	0.012	$2.5
seqrchquarry.com	$8,850	0.096	$850
asiatees.com	$36,000	0.017	$613
....	.....	.....	....
Total			$1.2 million

Wikipedia vs The Web

Revenue estimation:

- **Per website (all websites)**
- **Per website (highly reusing)**
- **Per Wikipedia web page**

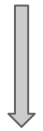


website	Monthly revenue	Percentage of reuse	Monthly Wikipedia value
pdxretro.com	$195	0.012	$2.5
seqrchquarry.com	$8,850	0.096	$850
asiatees.com	$36,000	0.017	$613
....	.....	.....	....
Total			$1.2 million

Wikipedia vs The Web

Revenue estimation:

- Per website (all websites)
- Per website (highly reusing)
- Per Wikipedia web page



website	Monthly revenue	Percentage of reuse	Monthly Wikipedia value
pdxretro.com	$195	0.012	$2.5
seqrchquarry.com	$8,850	0.096	$850
asiatees.com	$36,000	0.017	$613
....	.....	.....	....
Total			$1.2 million

Wikipedia vs The Web

Revenue estimation:

- Per website (all websites)
- Per website (highly reusing)
- Per Wikipedia web page

website	Monthly revenue	Percentage of reuse	Monthly Wikipedia value
pdxretro.com	$195	0.012	$2.5
seqrchquarry.com	$8,850	0.096	$850
asiatees.com	$36,000	0.017	$613
....	.....	.....	....
Total			$1.2 million

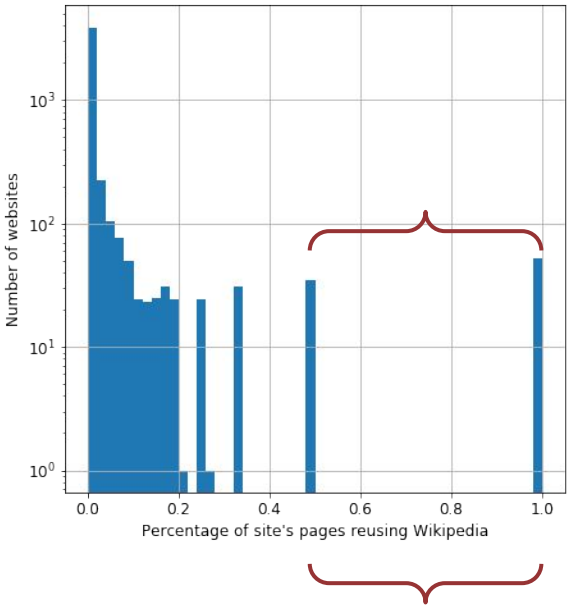
The rough estimate of monthly revenue of Wikipedia content

Bauhaus-Universität Weimar

Wikipedia vs The Web

Revenue estimation:

- **Per website (all websites)**
- **Per website (highly reusing)**
- **Per Wikipedia web page**



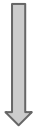
- Percentage of pages reusing Wikipedia >= 0.5
- 87 websites.
- Estimated monthly revenue: **$15k**

Wikipedia vs The Web

Revenue estimation:

- **Per website (all websites)**
- **Per website (highly reusing)**
- **Per Wikipedia web page**

Extracted from Wikipedia API



Reused Wikipedia page	Average page views	Average CPM	Average monthly revenue
Nuclear renaissance	645	$2.8	$1.806
Second Chechen War	34655	$2.8	$97
Enumerated powers	12858	$2.8	$36
....	.....	.....	....
Total			$900k

Wikipedia vs The Web

Revenue estimation:

- **Per website (all websites)**
- **Per website (highly reusing)**
- **Per Wikipedia web page**

Estimated from marketing reports

Reused Wikipedia page	Average page views	Average CPM	Average monthly revenue
Nuclear renaissance	645	$2.8	$1.806
Second Chechen War	34655	$2.8	$97
Enumerated powers	12858	$2.8	$36
....	.....	.....	....
Total			$900k

Wikipedia vs The Web

Revenue estimation:

- **Per website (all websites)**
- **Per website (highly reusing)**
- **Per Wikipedia web page**

Reused Wikipedia page	Average page views	Average CPM	Average monthly revenue
Nuclear renaissance	645	$2.8	$1.806
Second Chechen War	34655	$2.8	$97
Enumerated powers	12858	$2.8	$36
....	.....	.....	....
Total			$900k

Wikipedia vs The Web

Revenue estimation:

- **Per website (all websites)**
- **Per website (highly reusing)**
- **Per Wikipedia web page**

Reused Wikipedia page	Average page views	Average CPM	Average monthly revenue
Nuclear renaissance	645	$2.8	$1.806
Second Chechen War	34655	$2.8	$97
Enumerated powers	12858	$2.8	$36
....	.....	.....	....
Total			$900k

Wikipedia vs The Web

Monthly revenue:

Per Web sample	Number of reusing web pages	Revenue(per webpage)
59 million	15k	$900k
590 million	150k	$9 million

Wikipedia vs The Web

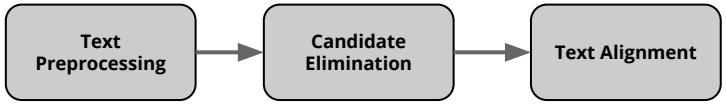
Monthly revenue:

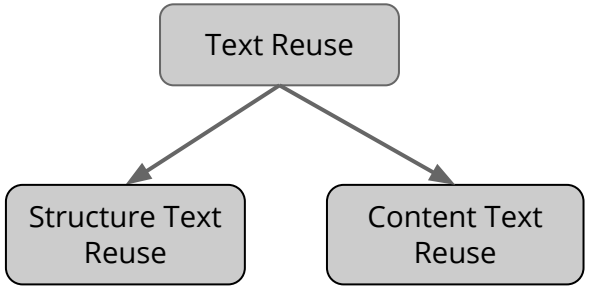
Per Web sample	Number of reusing web pages	Revenue(per webpage)
59 million	15k	$900k
590 million	150k	$9 million

Conclusion

Summary

- Pipeline for TR Extraction
- Text Reuse in Wikipedia
- Text Reuse between Wikipedia and the Web

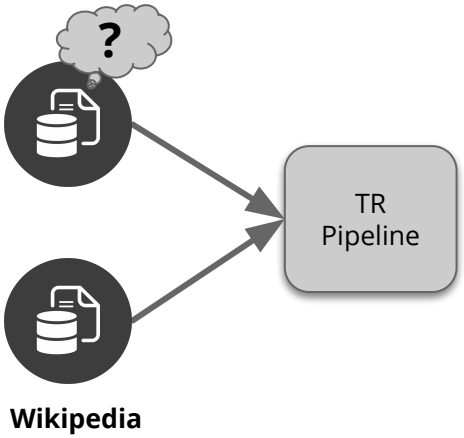




Per website (all websites)	Per website (highly reuse)	Per Webpage
$1.2 million	$15k	$900k

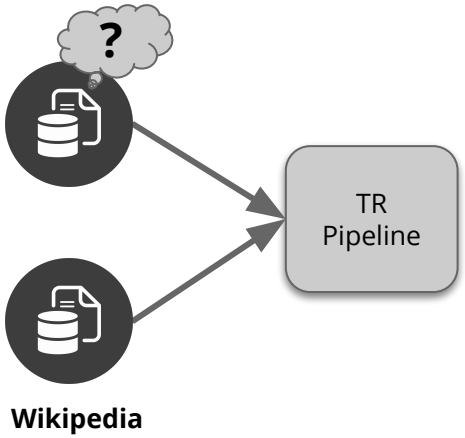
Future Work

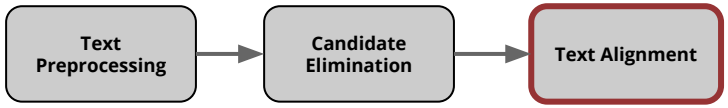
- Using the pipeline to extract and analyze TR between Wikipedia and the scientific community.
- Experiments on the Text Alignment subtask.
- Further analysis of the extracted Text Reuse cases.
- More accurate estimation on the monthly revenue generated by Wikipedia content.



Conclusion

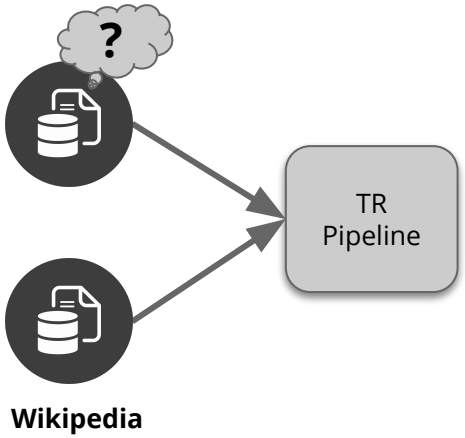
- Using the pipeline to extract and analyze TR between Wikipedia and the scientific community.
- Experiments on the Text Alignment subtask.
- Further analysis of the extracted Text Reuse cases.
- More accurate estimation on the monthly revenue generated by Wikipedia content.

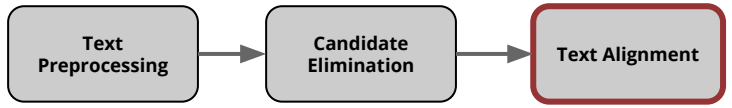


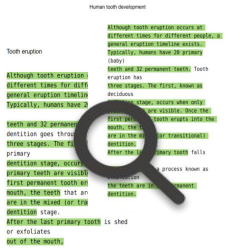


Conclusion

- Using the pipeline to extract and analyze TR between Wikipedia and the scientific community.
- Experiments on the Text Alignment subtask.
- Further analysis of the extracted Text Reuse cases.
- More accurate estimation on the monthly revenue generated by Wikipedia content.

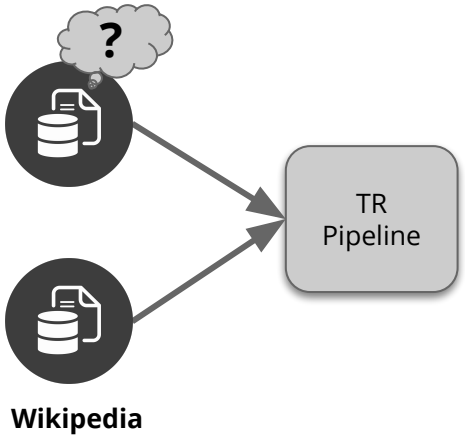


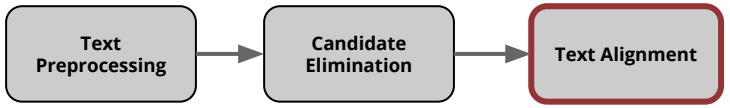


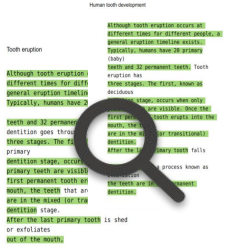


Conclusion

- Using the pipeline to extract and analyze TR between Wikipedia and the scientific community.
- Experiments on the Text Alignment subtask.
- Further analysis of the extracted Text Reuse cases.
- More accurate estimation on the monthly revenue generated by Wikipedia content.







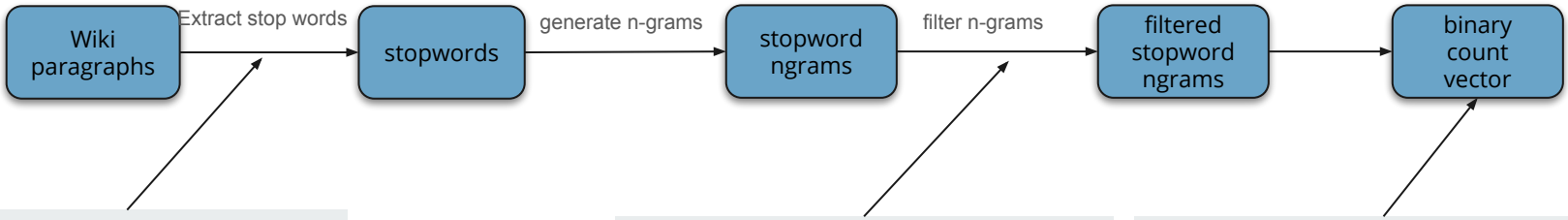
Backup Slides

- Candidate Elimination functions:

$$cand(s_i, d_i) = max(cosine\_similarity(p_i, p_j))) : p_i \in d_i, p_j \in d_i$$

$$cand(s_i, d_i) = max(\frac{count(shared\_ngrams(p_i, p_j)}{min(ngrams-count(p_i), ngrams-count(p_j)}) : p_i \in d_i, p_j \in d_i$$

- Stopwords N-grams procedure:



Top 50 frequent stopwords:

the, of, and, a, in, to,is, was, it, for, with, he, be, on, i, that, by, at, you, 's, are, not,his, this, from, but, had, which, she, they, or, an, were, we, their, been, has, have, will, would, her, there, can, all,as, if, who, what, said

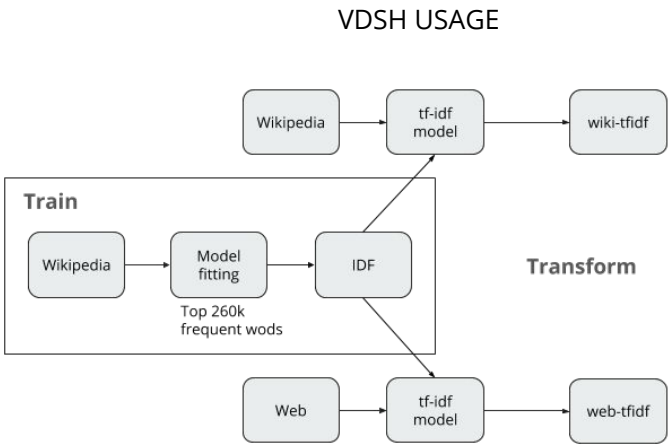
- Let C = *{the, of, and, a, in, to, 's}* stopwords that increases false positive.

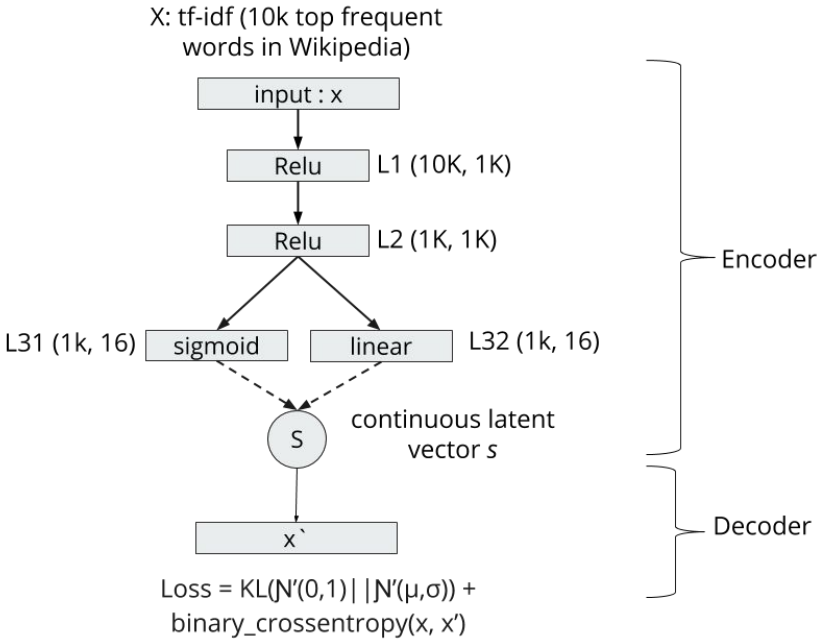
- X is accepted n-gram if:
  - It doesn't contain more than n-1 stopwords from C
  - The maximal sequence of stopwords belonging to C is less than n-2

- Binary count vector ignores the frequency in which a specific n-gram happened in a paragraph.

- We apply the scoring function on the binary count vector

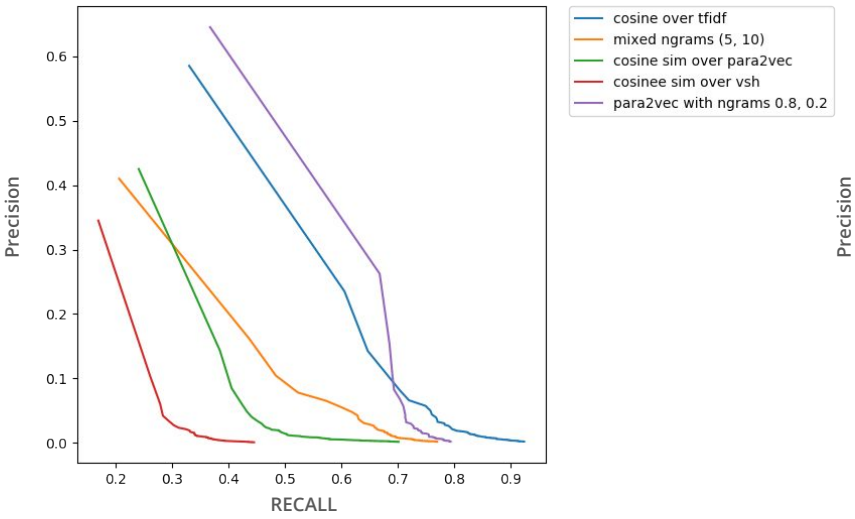
- VDSH explained:



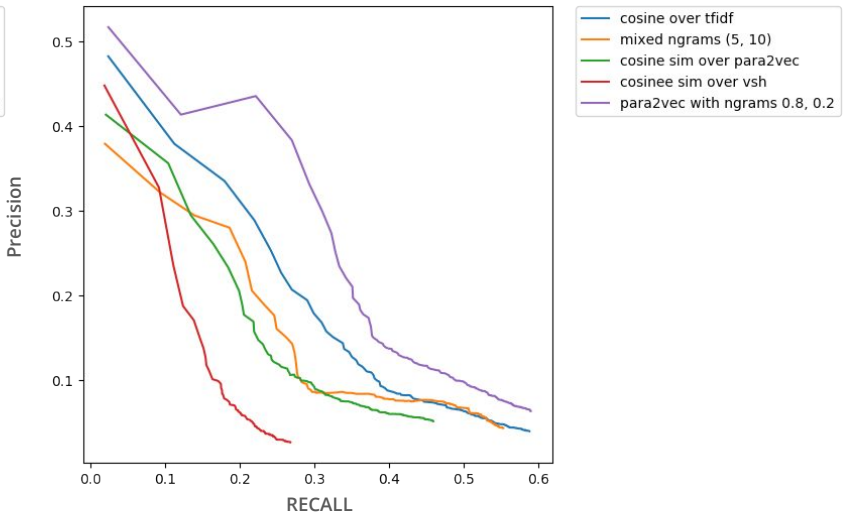


- Performance of candidacy functions on different thresholds:

Thresholds between (1 to 1000 and step of 5)



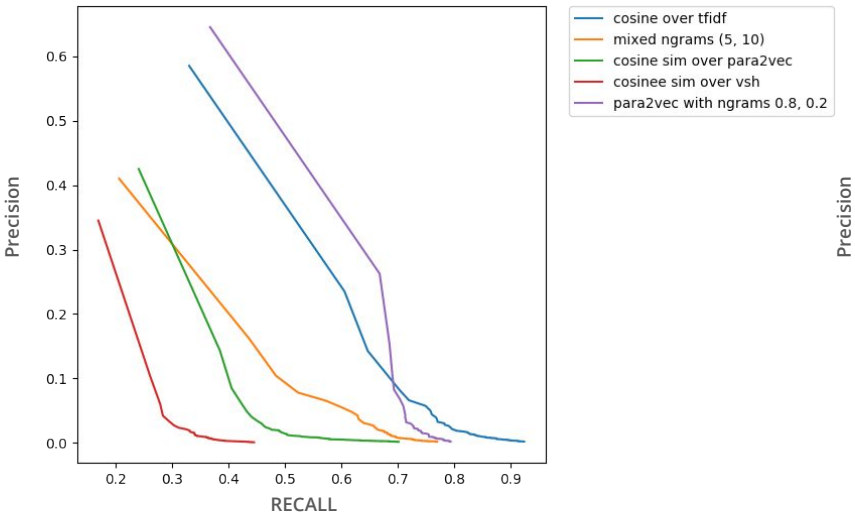
Documents from sample who have number of aligned docs <= 10



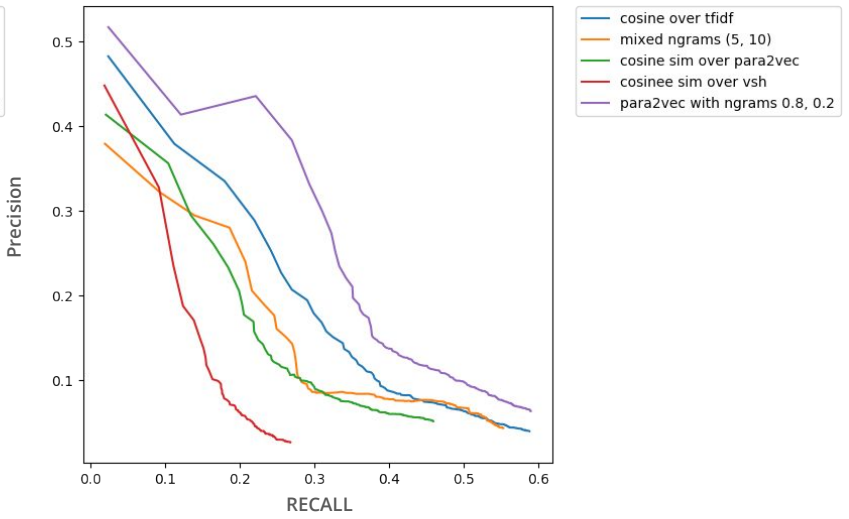
Documents from sample who have number of aligned docs > 10

- Performance of candidacy functions on different thresholds:

Thresholds between (1 to 1000 and step of 5)

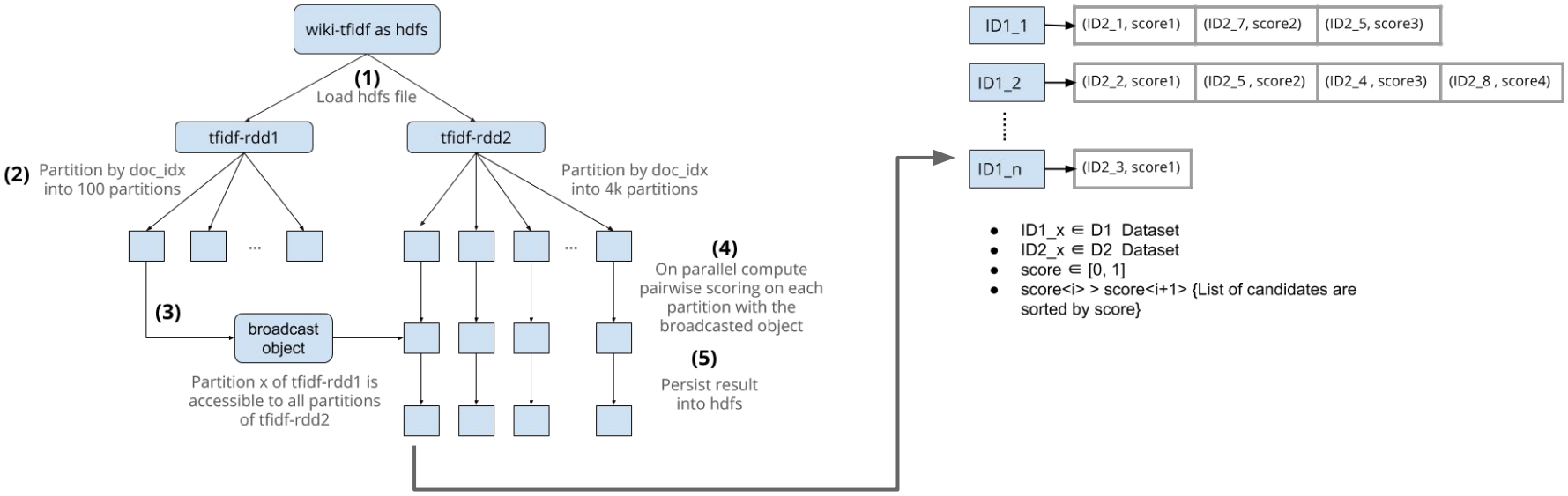


Documents from sample who have number of aligned docs <= 10

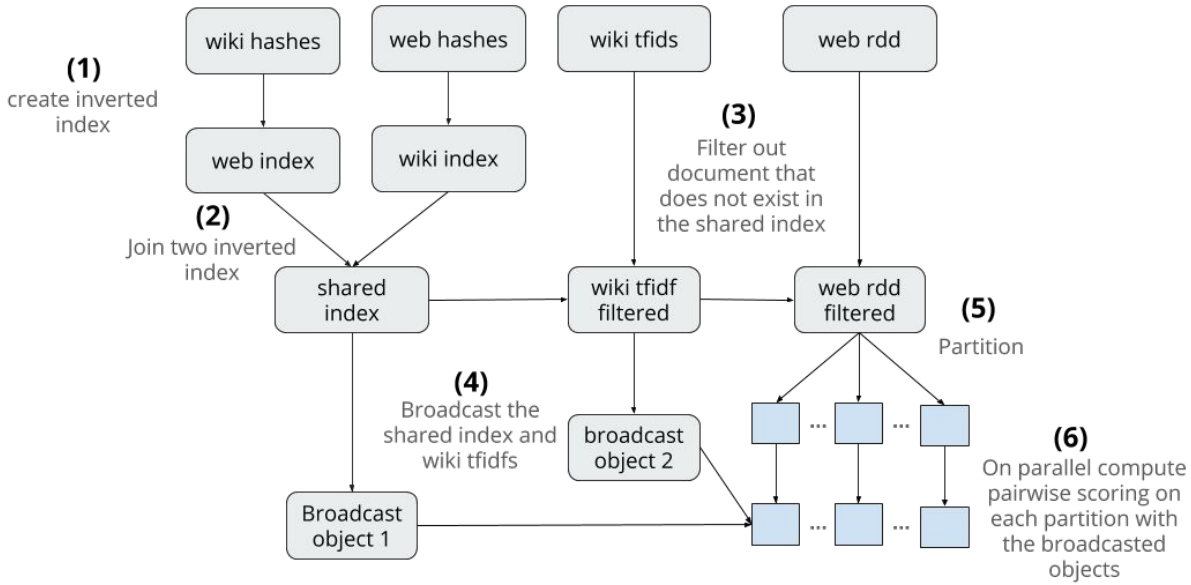


Documents from sample who have number of aligned docs > 10

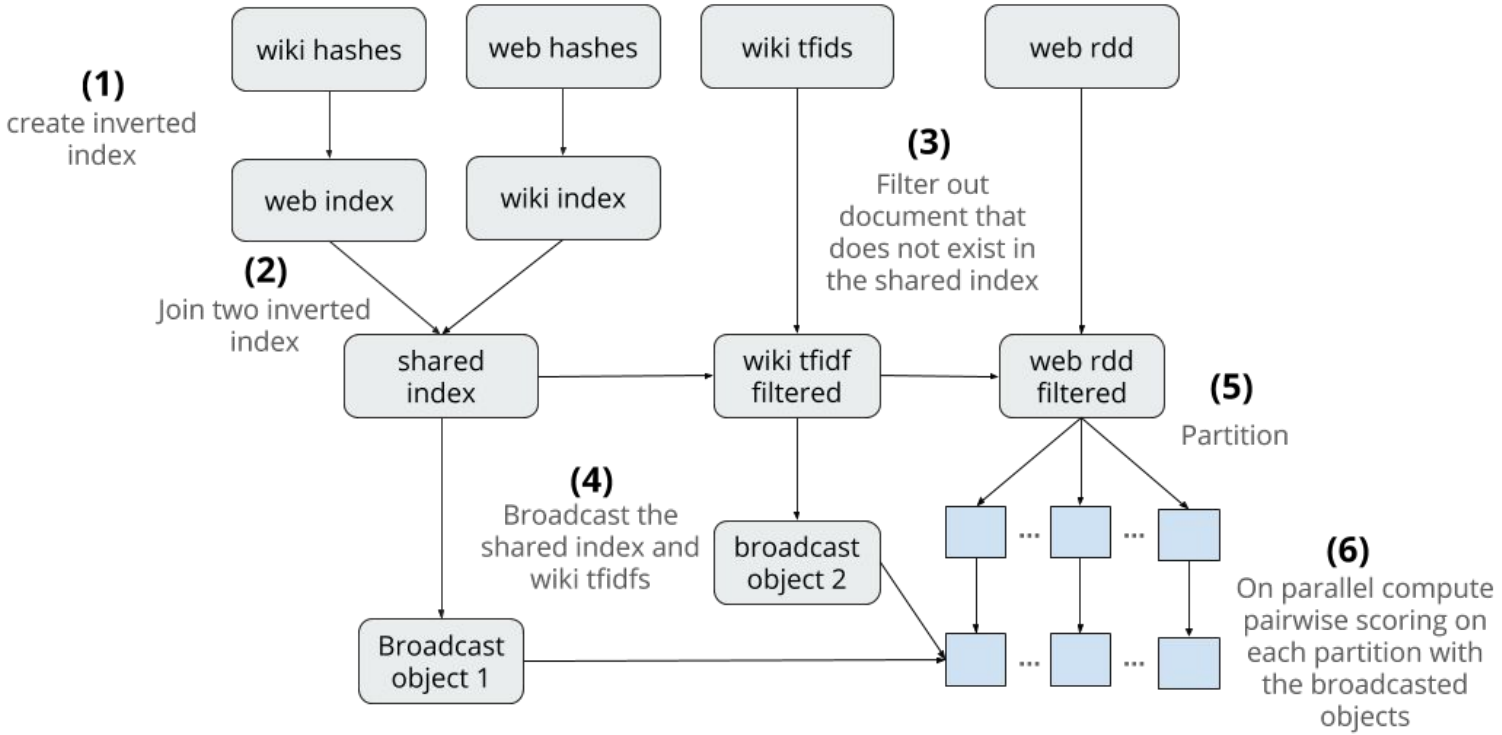
- Candidate Elimination procedure over the cluster:



- Hash based Candidate Elimination procedure over the cluster:



- Hash based Candidate Elimination procedure over the cluster:



- **Heuristics:**
  - *H1: ne_sim ∈ (0.5, 1.0] AND 10grams_sim > 0.5 AND (s_percent_reused < 0.5 or t_percent_reused < 0.5) => **content reuse** otherwise **structure reuse***
  - 6700 content reuse cases only
  - Validation on two random samples of size 100:

	Structure reuse	Content reuse
Sample1	100%	58%
Sample2 (Text1 or Text2 > 200)	100%	73%