Candidate Document Retrieval for Cross-Lingual Plagiarism Detection

Alexey Romanov, Anton Khritankov

Antiplagiat Research
Moscow Institute of Physics and Technology

Agenda

Types of Text Reuse

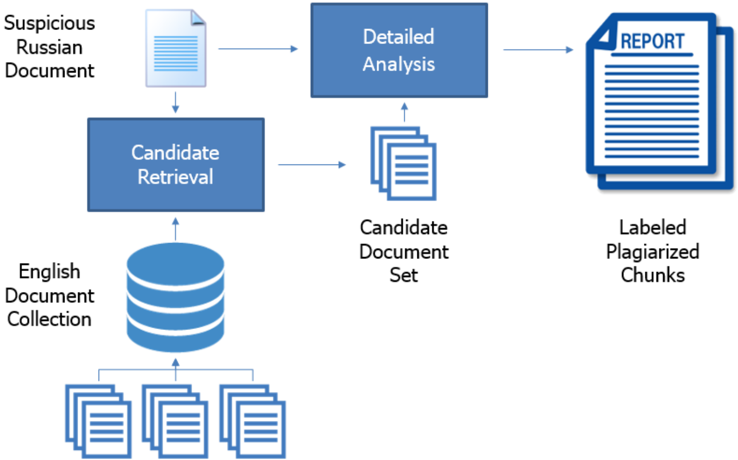
Text reuse (= “plagiarism”) can be classified into several categories:

- copying text “as is”
- text reuse with paraphrasing
  - *Mr. Dursley always sat with his back to the window in his office on the ninth floor.*
  - *Mr. Dursley always propped his back on the glass window on the ninth floor of the office.*
- cross-language plagiarism
  - *A cat was sitting on the table.*
  - *На столе сидела кошка.*

Cross-language plagiarism detection tackles challenges of two tasks:

- Machine translation
- Paraphrase detection

Cross-Language Plagiarism Detection Scheme



Problem Statement: Applied Version

- **Given:** English document collection
- **Given:** Suspicious Russian document
- **Relevance** between a suspicious document and a source document is amount of reused text normalized by the suspicious document length.
- **Task:**
  - Find candidate documents, which allegedly contain reused text from the suspicious document, in the collection.
  - Rank these documents according to their relevance values.
- Collection size: 10^6–10^9 documents.
- Candidate set size: 10–100 documents.

Problem Statement: Formal Version

- $E = \{e_1, \ldots, e_n\}$ — collection
- d — suspicious document
- $\varphi(d, e)$ — relevance
- $R_k(\varphi, d) = (e_{i_1}, \ldots, e_{i_k}) : \varphi(d, e_{i_1}) > \ldots > \varphi(d, e_{i_k})$, $\forall j : j \notin \{i_1, \ldots, i_k\} \rightarrow \varphi(d, e_{i_k}) \geq \varphi(d, e_j)$ — ranked plagiarism source list
- **Task:** find custom φ' approximating φ in the sense of preserving the ranking R_k
  - best case of φ':
    $\forall e_1, e_2 \in E \rightarrow \varphi(d, e_1) \geq \varphi(d, e_2) \Leftrightarrow \varphi'(d, e_1) \geq \varphi'(d, e_2)$

Problem Statement: Formal Version

- $Rel(d) = \{e \in E \mid \varphi(d, d') > 0\}$ — source set for d
- $R_k(\varphi', d)$ — ranking by φ'

For test set of Russian documents $D = \{d_1, \ldots, d_m\}$:

$$Q(k, \varphi', D, E) = \frac{1}{|D|} \sum_{d \in D} \frac{|R_k(\varphi', d) \cap Rel(d)|}{|Rel(d)|}$$

- Let $k = k_0 \geq \max_{d \in D} |Rel(d)|$, then $Q(k_0, \varphi', D, E) \leq 1$.
- **Task:** $Q(k_0, \varphi', D, E) \to \max_{\varphi'}$

Method Description

Proposed Method: Main Idea

- **Problem:** The majority of methods involve machine translation stage, which generates texts that differ too much from the sources of plagiarism.
  - *Having considered the dimensions next the policy analyst has to identify various indicators for each dimension.*
  - *Having considered the size of the following political analyst should identify the different indicators for each measurement.*
- **Idea:** Deal not with words but with word classes, which unite words and word forms that may be considered as translation of the same Russian phrases.
  - Obtain those word classes by clustering word embeddings on their cosine similarity.

Proposed Method: Word Embeddings

- **Word embeddings** (word2vec, GloVe etc.) are language modelling techniques of mapping words to vectors of real numbers.
- Vectors are learned by maximizing likelihood of certain words appearing in their contexts from the training data.
  - e.g. in word2vec skip-gram model:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_{t+j}|w_t)\to \max$$

    for some training sequence of words $w_1, \ldots, w_T$
  - words occurring in similar contexts get "cosine similar" vectors
  - $cos(v_1, v_2) = \frac{(v_1, v_2)}{||v_1||*||v_2||}$

Proposed Method: Word Embedding Clustering

Example queries for closest words to the GloVe model (trained on 42B token Common Crawl corpus)

```
model.most_similar('plagiarism')

[(u'dishonesty', 0.6062831878662109),
 (u'plagiarizing', 0.5256732106208801),
 (u'forgery', 0.5254061222076416),
 (u'plagarism', 0.5058314800262451),
 (u'plagiarized', 0.4934661090373993),
 (u'misconduct', 0.4911525249481201),
 (u'fraud', 0.48084795475006104),
 (u'turnitin', 0.47139039635658264),
 (u'cheating', 0.46822261810302734),
 (u'accusation', 0.46044373512268066)]
```

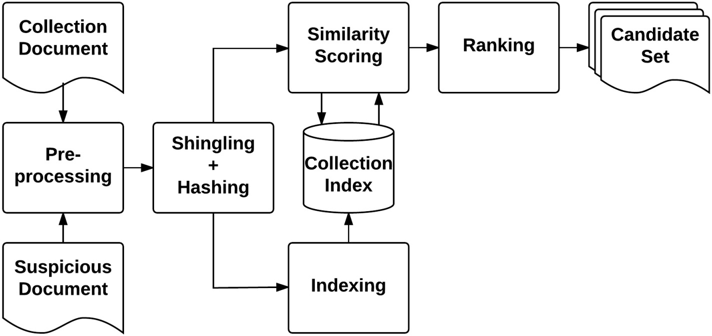
```
model.most_similar('detection')

[(u'detecting', 0.7272850275039673),
 (u'detector', 0.715851366519928),
 (u'detect', 0.6970372796058655),
 (u'detected', 0.6736979484558105),
 (u'detectors', 0.6247695684432983),
 (u'sensor', 0.6087626218795776),
 (u'detects', 0.6038689613342285),
 (u'monitoring', 0.598054051399231),
 (u'identification', 0.5846608877182007),
 (u'sensing', 0.582802414894104)]
```

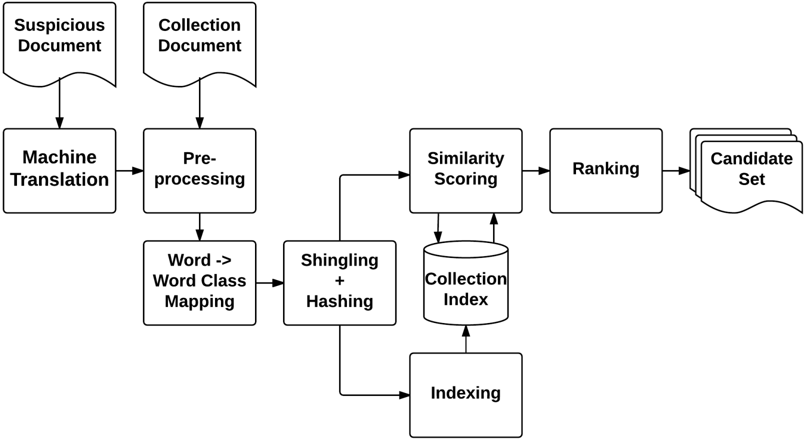
Cluster examples:

- *[beer, beers, brewing, ale, brew, brewery, pint, stout, guinness, ipa, brewed, lager, ales, brews, pints, cask]*
- *[survey, assessment, evaluation, evaluate, examine, assess, surveys, analyze, evaluating, assessments, examining, analyzing, assessing, questionnaire, evaluations, analyse, questionnaires, analysing]*
- *[brilliant, excellent, exceptional, finest, outstanding, superb, terrific]*

Proposed Method: Monolingual Shingle Search



Proposed Method: Cross-Lingual Shingle Search



Proposed Method: Implementation Details

- Most frequent hashes for the collection are not indexed
- Rare words are mapped to the single class
- Unknown words are removed
- Shingles — sorted overlapping word 4-grams
- $\varphi'(d, e) = \sum_{h \in H(d)} \frac{\mathbb{1}[h \in H(e)]}{|e' : h \in H(e')|}$
  - $H(d)$ — the set of document hashes
  - allows on-the-fly computation

Experiments

Experiment #1

- **Data:**
  - 1K sentence pairs from an English-Russian parallel corpus
  - machine translation of the Russian sentences into English
- **Methods:**
  - simple shingling (without mapping words to word classes)
  - shingling on word classes (proposed method)
- **Performance measures:**
  - q_{hash} — ratio of common hashes
  - q_{sent} — ratio of sentences where a common hash exists

Method	q_{hash}	q_{sent}
simple shingling	0.185	0.753
word-class shingling	0.221	0.796

Experiment #2: Data

- **PAN'11 corpus**:
  - 11K source documents + 11K suspicious documents
  - language: English
  - various plagiarism level:
    - length
    - limited number of sources
    - obfuscation: none / low / high
  - high obfuscation examples:
    - Christophe took her hands in his, kissed her, scolded her, spoke to her tenderly and roughly.
    - Christophe take her custody in his, had snog her, rebuke her, to her tenderly approximately.
  - low obfuscation is similar to machine translation errors, suitable for testing of the method

Experiment #2: Performance

- **Methods:** shingling on word classes (proposed method)
- **Performance measures:**

$$Q(k, \varphi', D, E) = \frac{1}{|D|} \sum_{d \in D} \frac{|R_k(\varphi', d) \cap Rel(d)|}{|Rel(d)|}$$

Obfuscation	$Q(k = 5)$	$Q(k = 10)$	$Q(k = 25)$
none	1.00	1.00	1.00
low	0.93	0.94	0.95
high	0.47	0.51	0.59

Experiments #3, #4

- **Data:**
  - 17K English papers on sociological topic
  - *[Experiment #3]* their machine-translated Russian versions
  - *[Experiment #4]* authentic Russian sociological papers with plagiarized chunks
- **Methods:**
  - CL-ESA (Potthast, M., Stein, B. (2011))
  - shingling on word classes
- **Performance measures:** $Q(k, \varphi', D, E)$

Experiment #3	$Q(k=1)$	$Q(k=5)$	$Q(k=10)$
CL-ESA	0.31	0.48	0.55
word-class shingling	1.00	1.00	1.00

Experiment #4	$Q(k=5)$	$Q(k=10)$	$Q(k=25)$
word-class shingling	0.93	0.95	0.96

Discussion

- Mapping of words to word classes enables smoothing of machine translation errors.
- CL-ESA (baseline) can be fooled by synonymic substitution and short plagiarized chunks.
- Errors of the proposed method result from:
  - plagiarized chunks of 1-2 sentences
  - archaic and rare words (*kissed / had snog*)
  - contextual synonyms (*hands / custody*)
  - synonyms used in different genres (*suffocation / asphyxiation)*

Conclusions

- Results of the study:
  - English word clustering
  - Method of candidate retrieval
  - Corpus of texts with cross-language text reuse
- The method can be applied to cross-language plagiarism detection task
- Further work may be aimed at:
  - enhancement of mapping to word classes
  - method parameter tuning
  - experiments on real-world data
  - scaling of the method to larger collections

Antiplagiat Research

Antiplagiat Research tackles the most challenging problems in the area of natural language processing and plagiarism detection.

- Development of advancing technology
- Propagation of scientific thought
- Unity of young talents from leading institutions
  - Moscow Phystech (MIPT)
  - Computing Centre of RAS
  - Moscow State University

We are looking for:

- talented researchers
- joint studies
- consulting & mentorship opportunities

Antiplagiat Research

Areas of our interest:

- Cross-Language Plagiarism
- Paraphrase Detection
- Machine-Generated Text Detection
- Automatic Text Categorization
- Intelligent Search and Topic Search
- Author Profiling
- Smart Evaluation of Research Papers

Thanks for you attention!

Questions / Comments?

Alexey Romanov
romanov@ap-team.ru