Plagiarism Candidate Retrieval Using Selective Query Formulation and Discriminative Query Scoring

Osama Haggag and Samhaa El-Beltagy
Center for Informatics Science,
Nile University,
Egypt

Nile University

Center for Informatics Science

Outline

- Introduction
- Problem Description
- Task Description
- Implementation
- Results
- Conclusion
- Future Work

Task Description

Task Description

- We are given a plagiarized dataset
  - Plagiarized from the ClueWeb09 corpus
  - There's little to no obfuscation
  - Some passages and headlines are not plagiarized
  - Documents are well written, and punctuated
  - Documents are organized into paragraphs focusing on certain **subtopics** related to the larger topic at hand

Task Description

- The goal is to:
  - Maximize and maintain a good balance in the retrieval performance
  - Minimize workload and runtime
- **The plan is to broaden the searching scope through topical segmentation**
- **While introducing some form of search control in utilizing the queries**
  - It would be favorable to score queries that haven't been used yet against already downloaded documents
- The core of the problem is document downloads
  - Downloading irrelevant documents leads to more irrelevance
  - Downloading relevant documents minimizes the search effort and sharpens precision

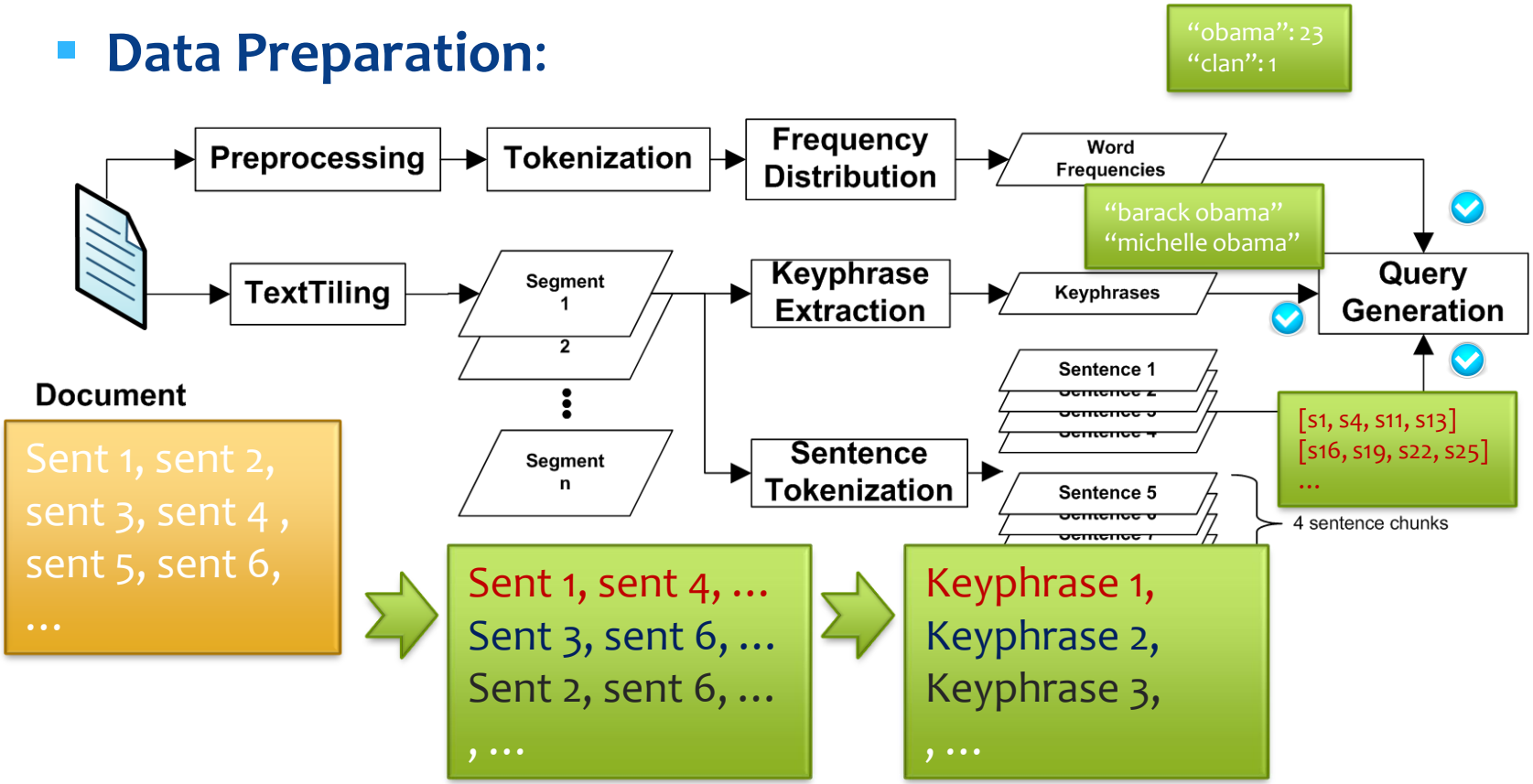
Implementation

Implementation

- The slight obfuscation was disregarded due its insignificance
- ChatNoir is the search engine of choice
- The system is made up of a number of phases
  - Data preparation
  - Query formulation
  - Searching
- Tuning the parameters

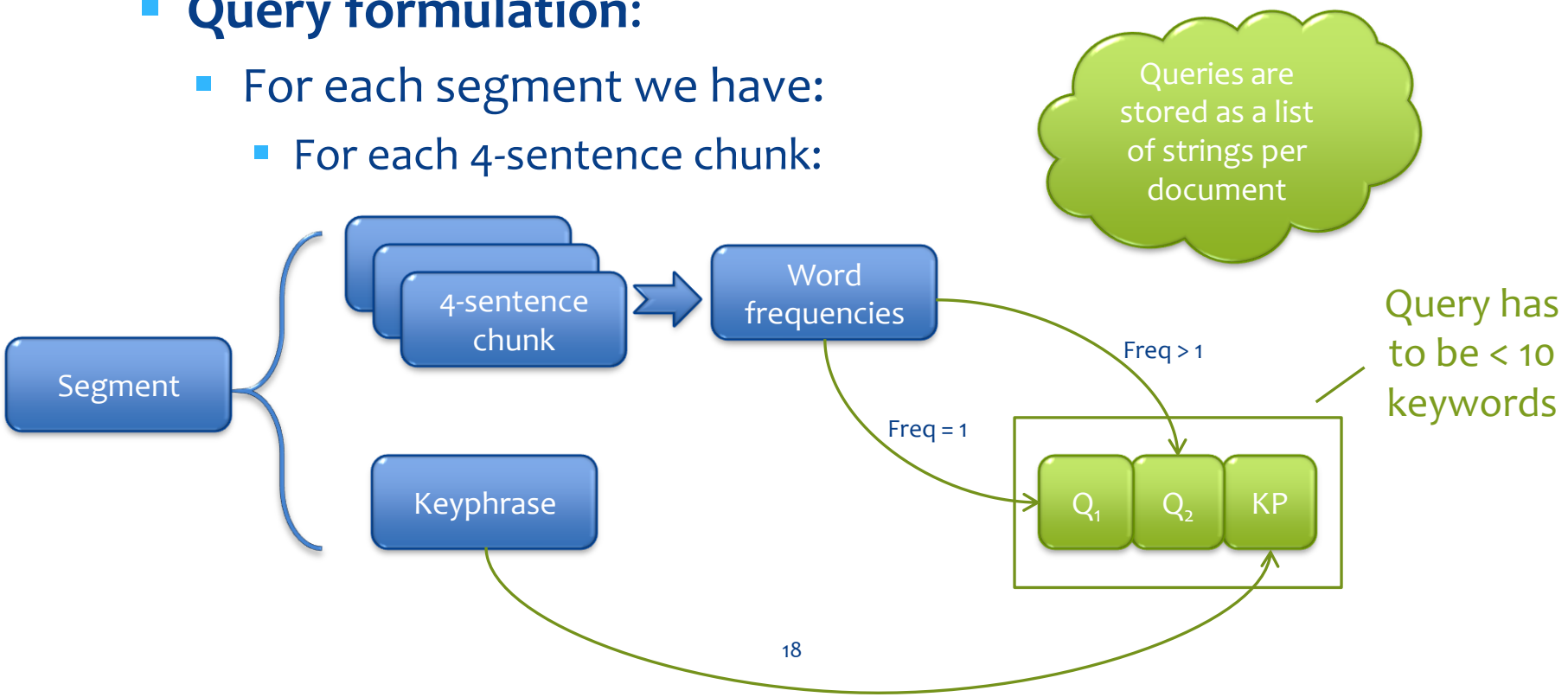
Implementation

- **Data Preparation:**



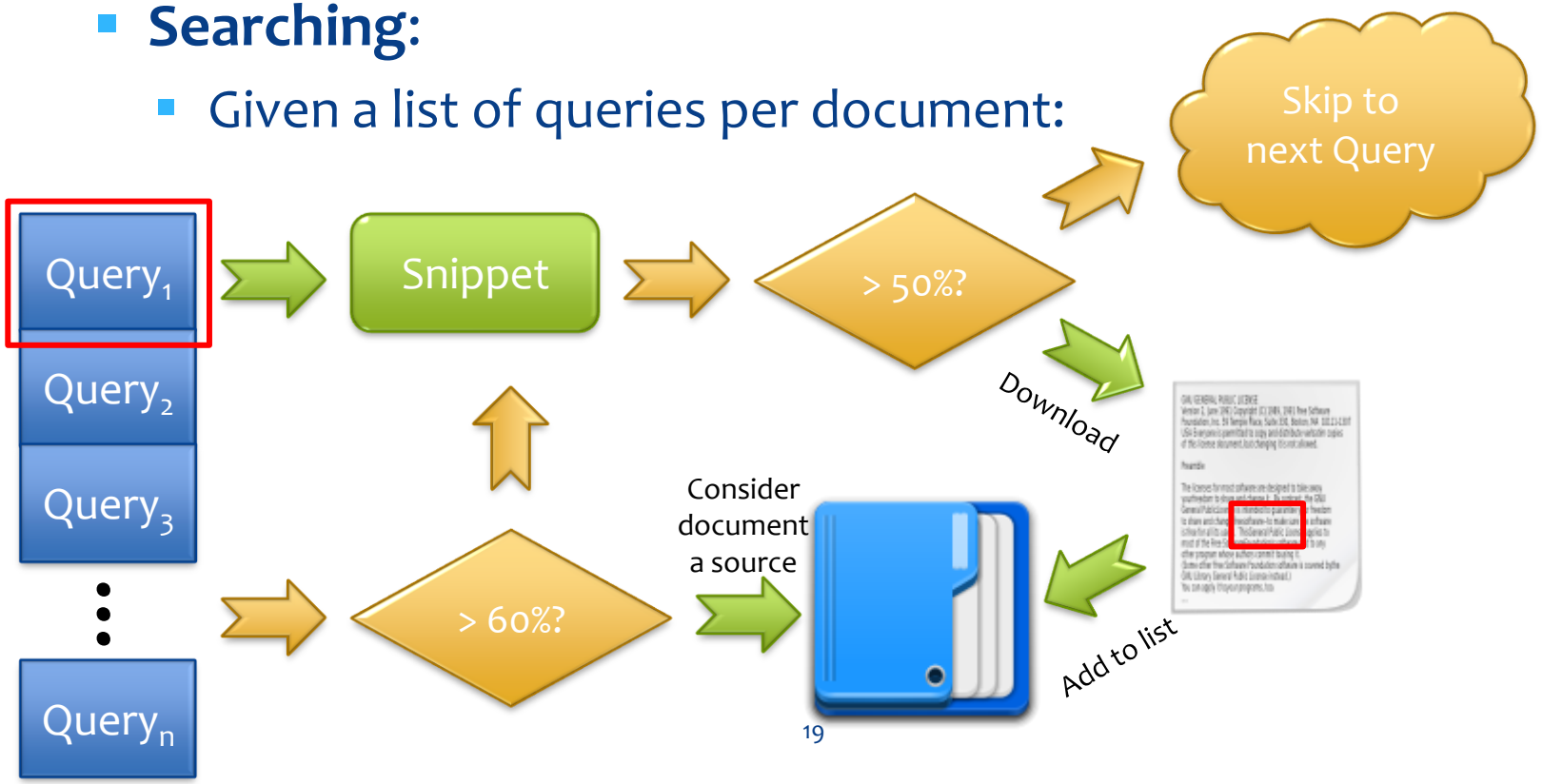
Implementation

- **Query formulation:**
  - For each segment we have:
    - For each 4-sentence chunk:



Implementation

- **Searching:**
  - Given a list of queries per document:



Implementation

- **Tuning the parameters:**
  - The system has a number of parameters that need tuning
  - Due to the time cost of an experiment over the dataset, difficult to optimize by iteration over combinations
  - We use human intuition, common sense, and a small number of experiments to determine values that are good enough, but not necessarily optimal

Implementation

- **Tuning the parameters (in processing):**
  - **TextTiling parameters:**
    - Control over size of subdocuments
    - Tuning for a large number of segments of small size gives higher recall
    - Tuning for a small number of large topics is best for both precision and recall

Implementation

- **Tuning the parameters (in processing):**
  - **Sentence chunk size selection:**
    - A chunk size of 1, gives better recall at loss of precision
    - A chunk size of 4 is determined to do best
  - **Frequency threshold:**
    - Identifies the "unique" words in the query
    - The threshold of 1 is chosen after running experiments

Implementation

- **Tuning the parameters (for search):**
  - **Number of results returned:**
    - First result is often the most relevant one
  - **Query vs. Snippet score:**
    - A score of 50% filtered search results nicely
    - Less meant higher recall, more meant less recall without equivalent improvement in precision

Implementation

- **Tuning the parameters (for search):**
  - **Query vs. Candidate Document score:**
    - Same rationale as scoring against snippets
    - 60% a relatively good filter
    - Higher values are better for recall
- Refer to Tables 1,2,3 on page 6 in the paper for details

Results

Results

- Our system was evaluated using the measures set by PAN'13
- The system is determined to be one of the top three systems at PAN'13

	Retrieval Performance			Workload		1st Detection		No Detection	Runtime
	F1	*Prec*	*Recall*	*Qrs*	*Dlds*	*Qrs*	*Dlds*		
Haggag	0.44	**0.63**	0.38	**32.04**	**5.93**	8.92	**1.47**	9	**9162471**
Williams	**0.47**	0.55	**0.50**	116.4	14.05	17.59	2.45	**5**	69781436
Lee	0.35	0.50	0.33	44.04	11.16	**7.74**	1.72	15	18628376

Conclusion

Conclusion

- We have a system that can retrieve possible plagiarism sources with competitive performance at minimal workload
- This is done through careful formulation, and discriminative elimination of queries
- The system employs two algorithms
  - TextTiling: topical segmentation – Marti A. Hearst
  - KPMiner: keyphrase extraction – Samhaa R. El-Beltagy

Future Work

- There is room for improvement on the current system
  - Optimize the parameters
  - Make use of ChatNoir's advanced search functions
- Investigate more about obfuscation
- More intelligence in the scoring functions
- The code to our implementation available on git-hub, under the MIT license


THANK YOU!