Automated Story Illustrator

Anshita Talsania
Charotar University of Science and Technology (Charusat),
Anand,
India.
anshita.talsania@gmail.com

Sandip Modha
L.D.R.P. College,
Gandhinagar,
India.
sjmodha@gmail.com

Hardik Joshi
Gujarat University,
Ahmedabad,
India.
joshee@acm.org

Amit Ganatra
Charotar University of Science and Technology (Charusat),
Anand,
India.
amitganatra.ce@charusat.ac.in

ABSTRACT

The Automated Story Illustration is a task under FIRE 2015 to be organized in DAIICT, Gandhinagar. The participants are required illustrate stories automatically by retrieving a set of images from an image dataset and if it is the case, identifying which concepts and events in the text should be illustrated. This paper overviews the task, the approach- the model and the tool used to carry out the task.

Keywords

Automated Story Illustrator, Illustrated story.

1. INTRODUCTION

The task “Automated Story Illustration” requires to illustrate stories automatically by retrieving set of images from the image dataset. The key tasks involved to are:

a. Identify concepts and events in the text that should be illustrated (annotations).

b. Selecting best illustration from the image dataset for that particular concept/event.

In the FIRE task, participants are provided with multiple children’s short stories which need to be illustrated using the ImageCLEF Wikipedia Image Retrieval dataset. The story text as well as the important entities and events that need illustration [1] in it are provided. The objective is to provide one ranked list of images corresponding to each important entity and event in a story.

The need of this research stems from the fact that we often forget what we read few sentences before. Our reading memory is affected due to boredom, lack of attention or distraction. Studies [2] suggest creating visual illustrations can improve reading memory therefore going a long way in helping children and elder people who often face reading memory problems.

2. DATA

The task uses two different data components: one is an image dataset containing all possible images available for illustration and another one is the children's short stories that need to be illustrated. In this task, ImageCLEF Wikipedia Image Retrieval 2010 is used as the image dataset. This dataset consists of 237,434 images along with their captions metadata. Captions are available in English, French and/or German. Secondly, the stories that need to be illustrated are all children’s short stories. Additionally annotations are provided for each story to indicate what portion of the story needs to be illustrated. Annotations are in form important entities (nouns or noun phrases) and events (a combination of entities and verbs or verb phrases) that need illustration in a story.

3. APPROACH

For creating the story illustration, primarily we need to perform information retrieval – query important passages of the story and retrieve the corresponding image representation available from the image dataset provided i.e perform two main tasks:

a. Indexing -Mapping of terms (basic indexed units) to documents in a corpus

b. Retrieval - Generation of results due to a query (information need)

To perform these tasks, several open source tools are available. These tools differ on the grounds of their indexing and retrieval models used. We choose the *Terrier*tool [3] as it is ideal for performing information retrieval experiments. Terrier can index large corpus of documents with multiple indexing strategies. Additionally it is highly effective providing support of state of art retrieval approaches like DFR and BM25.

3.1 Indexing

Indexing using the Terrier tool is performed on the ImageCLEF dataset provided. The tool utilizes entire image caption metadata represented in form of XML and then performs indexing using configuration set within the tool.

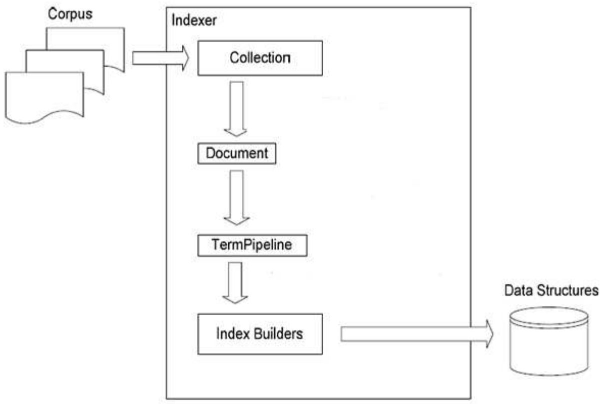

Figure 1: Indexing Architecture

Figure 1: outlines the indexing process in Terrier

The corpus data (*ImageCLEF*) is parsed in TREC format and that data forms the collection. A Collection object extracts the raw content of each individual document and hands it in to a Document object. The Document object then removes any unwanted content (e.g., from a particular document tag) and gives the resulting text to a Tokeniser object. Unwanted content is removed through the TermPipeline, which transforms the terms removing stopwords (high frequency terms) and stemming (prefix, suffix removal) [4]. Finally, the tokeniser object converts the text into a stream of tokens that represent the content of the document. The entire iterations of terms and is building of index is performed by *BasicIndexer*(default indexer).

We get output in form 10177882 tokens from the corpus data of 237434 images. Result of indexing is depicted below:

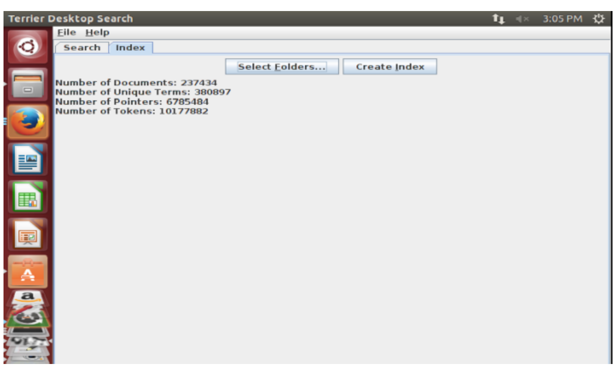

Figure-2: Screenshot of result of indexing performed in terrier tool

3.2 Retrieval

After performing indexing, we can now initiate retrieval process using different queries. These queries are in natural language and denote the text from the story that needs to be illustrated. We use the event description of a story as a search query. These event descriptions are provided as a XML data in the task with respective manual annotations. Each event from each story is used as query. Figure-3 provides an outline of the retrieval process in Terrier:

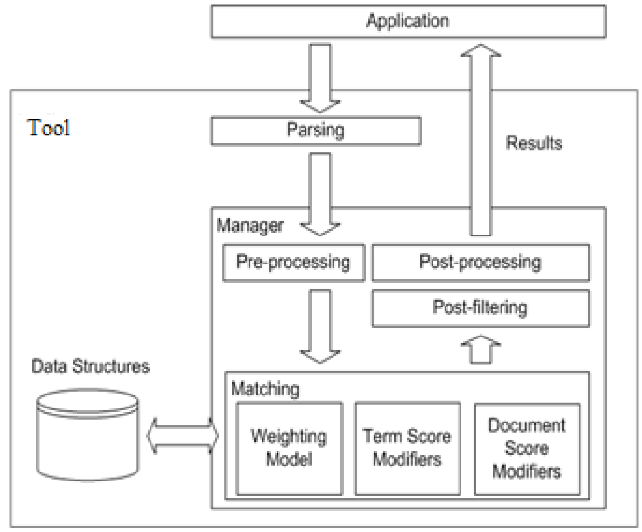

Figure 3: Retrieval Architecture

The input query is parsed initially post which it enters in Pre-processing – entering it into same configured TermPipeline. The query is then handed to matching component. Weighting Model is instantiated (DFR model is used) and document scores for the query are then computed. To improve the scores, query moves to post processing e.g query expansion [3] - taking the top most informative terms from the top-ranked documents of the query, and adding these new related terms into the query. The processed query is assigned scores by matching component. Post filtering is the final step in Terrier's retrieval process, where a series of filters can remove already retrieved documents, which do not satisfy a given condition.

Figure-4: depicts retrieval results of the query. In the screenshot we observe that Image ID 232878matches query with a highest score of 10.7788.

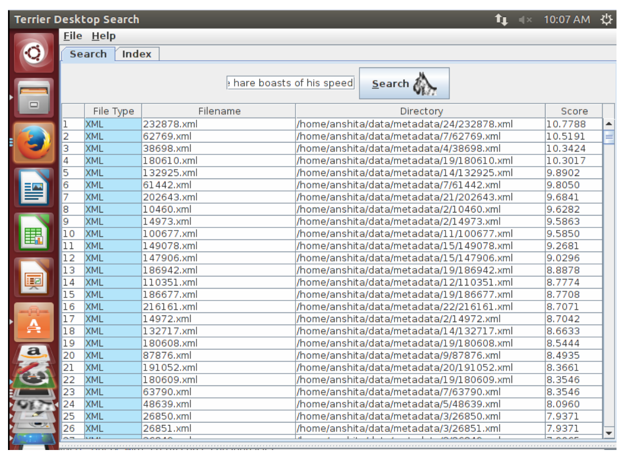

Figure-4: Screenshot of retrieval results of the query in terrier tool

After searching every query, run files (output) are compiled using the scores of search query and retrieved image id.

4. EVALUATION AND RESULTS

Evaluation is conducted on the run files using standard trec_eval tools. Precision-at-K (P@K) and mean average precision (MAP) scores are evaluated. Each important entity or event in a story will have a relevance list associated with it. P@K and MAP for each annotation are computed against these relevance scores.

There were a total of two groups participating and four system submissions with the result shown below in Table-1.

Table-1 RESULT

Run Name	num_ret	num_rel	num_rel_ret	MAP	MRR	B-pref	P@5
TFIDF-1	6405	2068	255	**0.0107**	0.1245	**0.1241**	0.0636
cguj-run1	92	2068	16	0.0047	**0.3708**	0.0074	0.1273
cguj-run2	95	2068	20	0.0053	0.2997	0.0095	**0.1545**
cguj-run3	100	2068	13	0.0030	0.2504	0.0065	0.0909

A highly effective information retrieval is one with high recall and precision i.e. retrieve as many relevant documents as possible and as few non-relevant documents as possible. The results of cguj-run-2 file had 0.1545 precision.

5. CONCLUSION

Automated Story Illustrator task is first time released in FIRE. The research can go a long way in illustrating short stories especially for children as well help improve reading memory. Lots of further work needs to be carried out in the task to improve the effectiveness of the result like modifying scores using different algorithms, improve the manual annotations and modify the queries or improving the indexing.

6. ACKNOWLEDGEMENTS

We are grateful to Dr. Debasis Ganguly and Mr. Iacer Calixto for his guidance throughout of this task. Additionally we would also like to thank FIRE2015 for opportunity to work under this task and facilitating the process.

7. REFERENCES

[1] Diogo Delgado , Joao Magalhaes and Nuno Correia. 2010. Assisted News Reading with Automated Illustrations. ACM. DOI= http://dl.acm.org/citation.cfm?id=1874311.

[2] Delgado, D. 2010. Automated illustration of news stories. IEEE.

[3] Michel Beigbeder, Wray Buntine and Wai Gen Yee. 2006. Terrier: A High Performance and Scalable Information Retrieval Platform. OSIR

[4] William B. Frakes, Ricardo Baeza-Yates, 1992. Book: Stemming Algorithms. PEARSON Education.