Matching documents and summaries using key-concepts

Sara Tonelli and Emanuele Pianta
Fondazione Bruno Kessler, Trento (Italy)

Overview

- Motivation of key-concept based approach
- Workflow description
- Experimental setup and evaluation
- Conclusions & Future work

Introduction to Key-concept Extraction

- What are key-concepts?

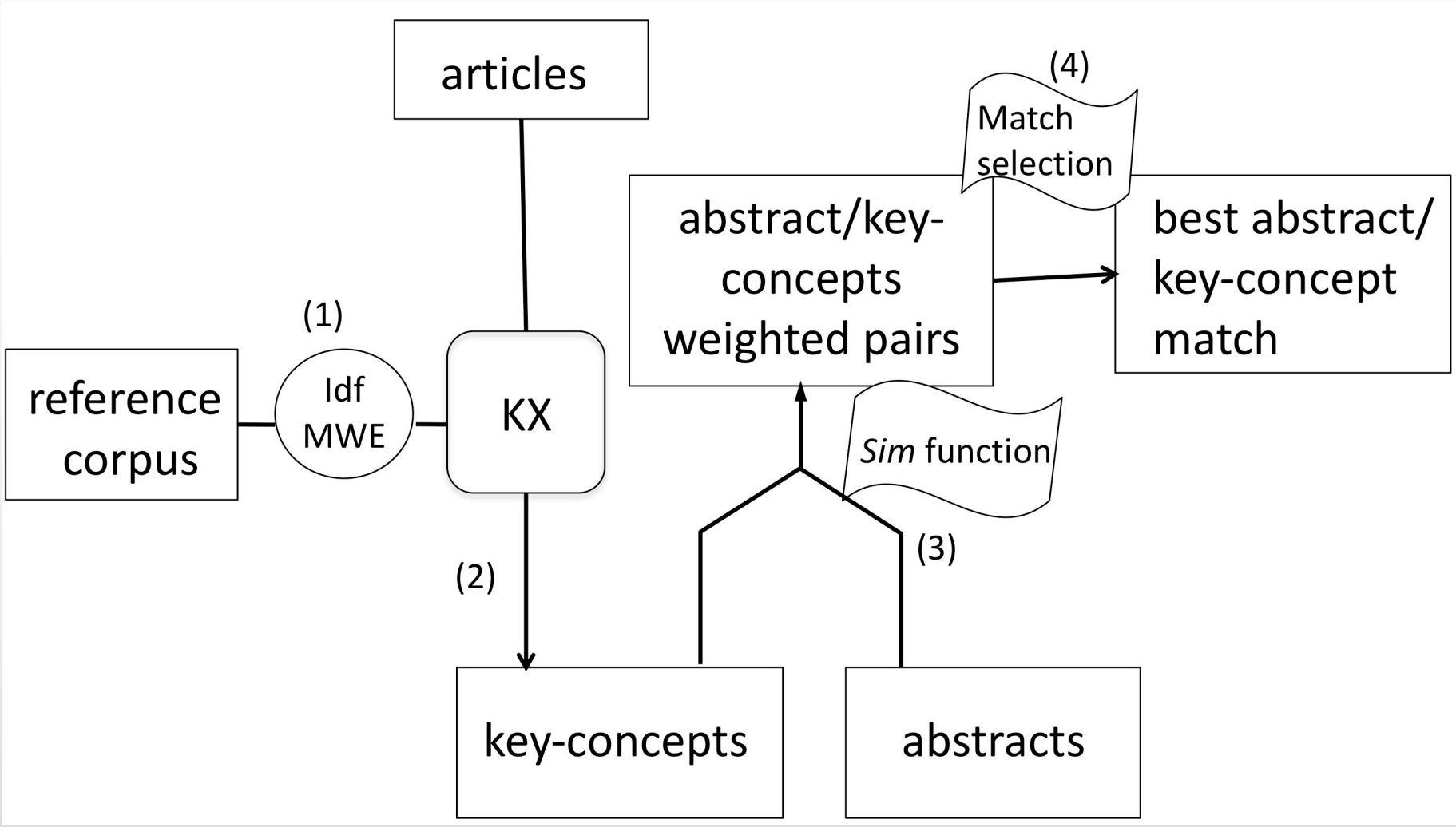
  Expressions (single or multi-words) describing the *most important concepts of a document*. Approximate characterization of the content of a text

- Why are they useful for NLP?

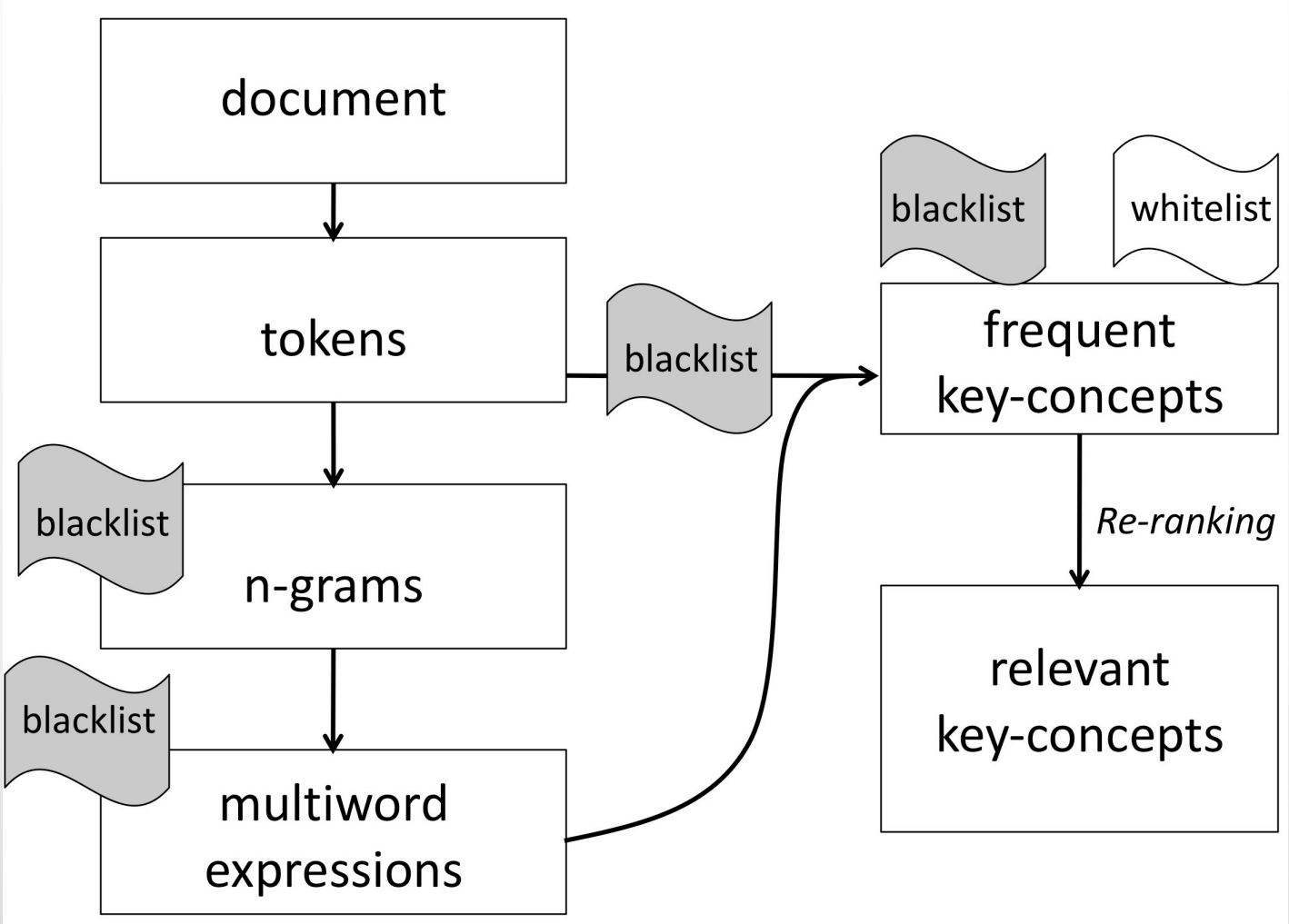
  Document indexing for quick topic search, document clustering, etc.

Our idea:
Use key-concepts to compute a similarity score between articles and abstracts

General matching workflow



Overview of KX key-concept extractor



Configurable parameters for re-ranking

Parameter list:

- Key-concept IDF
- Key-concept length

Ex. expression verbale = 6 → *6 x 2 = 12*

expression verbale des émotions = 5 → *5 x 4 = 20*

- Position of first occurrence
- Shorter concept subsumption / Longer concept boosting

Ex. expression verbale = 4 → *0*

expression verbale des émotions = 6 → *6 + 4 = 10*

Similarity score assignment

After extracting key-concepts K from each document, we assign a *similarity score* between K and each abstract A based on *coherence* and *completeness*:

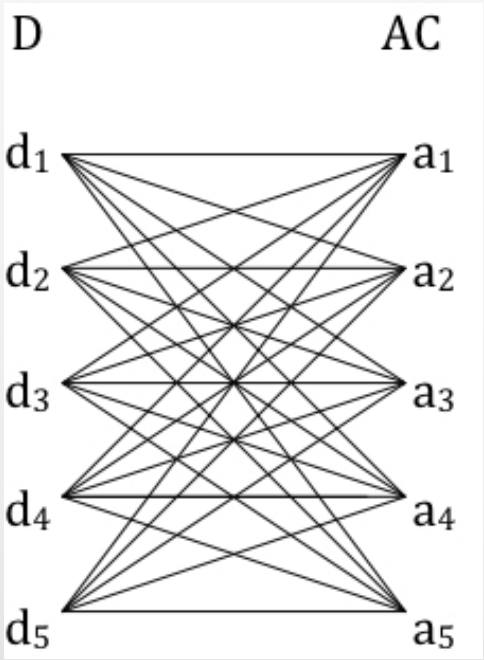
$Sim\_{Precision}$: Percentage of K contained in A

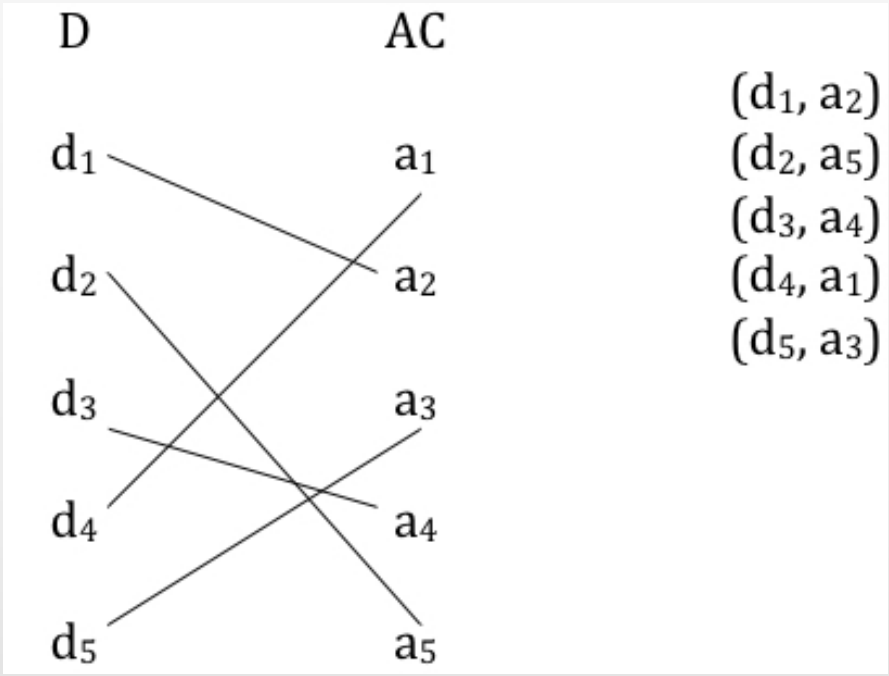
Sim_{Recall}: Percentage of tokens in A, excluding stopwords, that occur also in K

$$Sim\_{F1} = \frac{2 \times Sim\_{Precision} \times Sim\_{Recall}}{Sim\_{Precision} + Sim\_{Recall}}$$

Final selection of best match

From a complete bipartite graph to a perfect matching between (d_i,a_i) pairs





D = Documents
AC = Abstract collection

Final selection of best match

Strategy: find the set of edges in the graph that achieves the highest weight at *local* level

Example: find best match between $\{d_1, d_2, d_3\}$ and $\{a_1, a_2, a_3\}$

	a1	a2	a3
d1	3	0	8
d2	6	4	3
d3	5	2	7

Experimental Setup: Results on training set

	Full articles	**Short articles**
N. of key-concepts for each document	60	30
Multiply relevance by key-concept length	Yes	No
Consider position of first occurrence	Yes	No
Shorter concept subsumption	Yes	Yes
Longer concept boosting	Yes	Yes
Journal-based match	No	No
Precision Recall **F1**	P 0.976 R 0.966 **F1 0.971**	P 0.943 R 0.940 **F1 0.941**

Experimental Results on test set

	Full articles	Short articles
Setting as in previous table	0.960	0.934
+ corpus Idf	0.975	**0.964**
+ journal-based match	**0.990**	0.964

No linguistic pre-processing
A standalone system for key-concept extraction was used
No real supervision, only a development phase
Approach can be extended as it is to other languages

Conclusions

- Presentation of a workflow to match documents and summaries based on key-concepts
- Simple *similarity measure* inspired by F1
- Closing remark:

  KX freely available with *English* and *Italian* linguistic filters included. A version working also on *French*, *Finnish* and *Swedish* can be obtained without linguistic component. Feel free to contact us! satonelli@fbk.eu pianta@fbk.eu

Thank you!