Improving Keyphrase Extraction from Web News by Exploiting Comments Information

Zhunchen Luo, Jintao Tang and Ting Wang

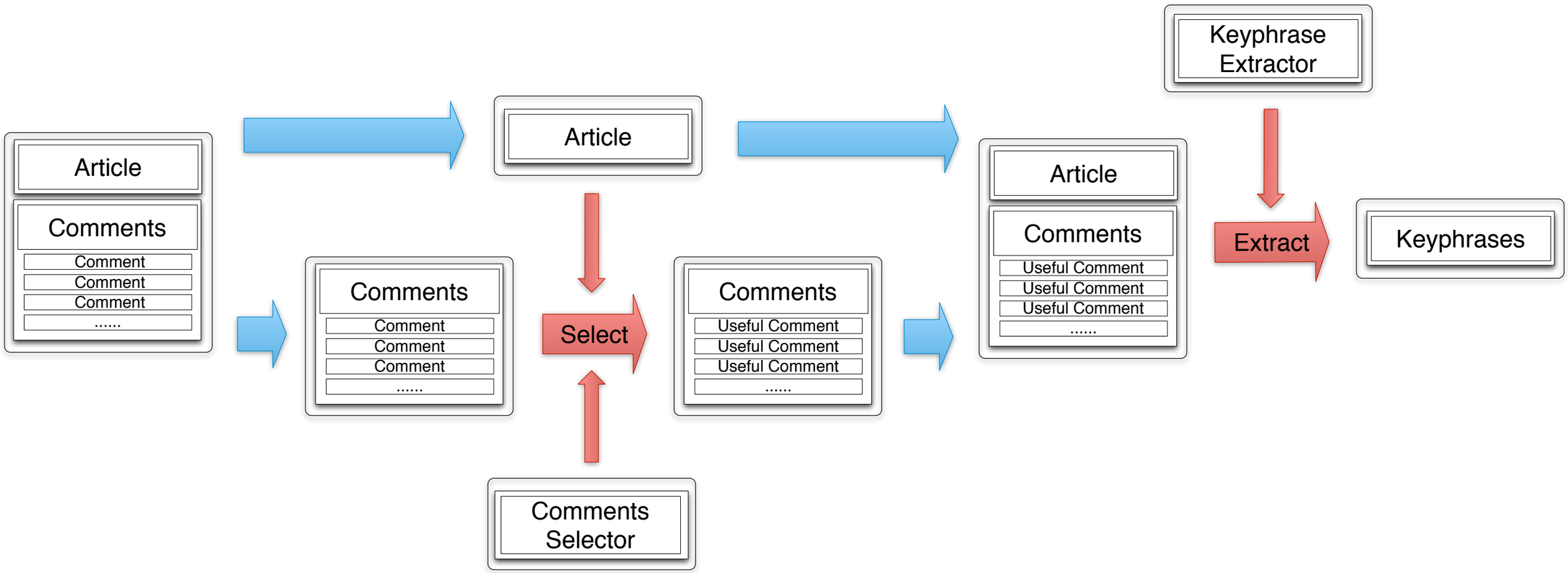
Keyphrase Extraction from Web News

- Most existing work treats each web news as an isolated document.
- Many web news sites provide various social tools for people to post comments.
- Goal: improving keyphrases extraction from web news by exploiting comments information.

Comments

- **Motivation**
  - Highly related to the web news documents (Topic Relevance).
  - Naturally bound with web news (Obtain Easily).
- **Challenge**
  - The quality of comments; How to choose useful comments?
  - Useful comments: providing additional information for readers to better understand the content of the article.

Framework of Keyphrase Extraction from Web News



Selecting Useful Comments

- **Similar Comments Selector**: selecting comments which are similar to the original article.
  - *Multiple documents within an appropriate cluster context usually have mutual influences and contain useful clues (Wan and Xiao, AAAI-08).*
  - Example: two documents related topic "earthquake"; common phrases: "earthquake" and "victim".

Selecting Useful Comments

- **Helpfulness Comments Selector**: selecting comments which have high helpfulness score.
  - Web news sites allow readers to vote for the helpfulness of each comment and provide a function *f* to assess the quality.

$$f = \frac{\mathrm{Num}_{thumbup}}{\mathrm{Num}_{thumbup} + \mathrm{Num}_{thumbdown}}$$

Selecting Useful Comments

- **KeyInfo Comments Selector**: selecting comments which include more keyphrases.
  - If a shorter comment contains more keyphrases, it is more likely to improve the performance of keyphrases extraction from the web news when integrating it into the original article.

$$k = \frac{\mathrm{Num}_{keyphrase}}{\mathrm{Num}_{word}}$$

Selecting Useful Comments

- **KeyInfo Comments Selector**: selecting comments which include more keyphrases.
  - If a shorter comment contains more keyphrases, it is more likely to improve the performance of keyphrases extraction from the web news when integrating it into the original article.

$$k = \frac{\mathrm{Num}_{keyphrase}}{\mathrm{Num}_{word}}$$

Selecting Useful Comments

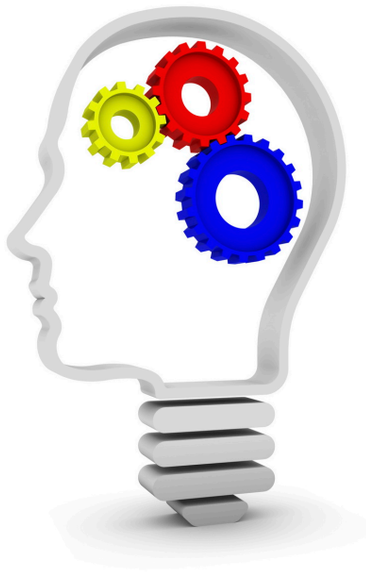
- **KeyInfo Comments Selector**: selecting comments which include more keyphrases.
  - If a shorter comment contains more keyphrases, it is more likely to improve the performance of keyphrases extraction from the web news when integrating it into the original article.

$$k = \frac{\mathrm{Num}_{keyphrase}}{\mathrm{Num}_{word}}$$

How to estimate it?

KeyInfo Comments Selector

- Predict the value of ***Num*$_{keyphrase}$**: a regression-learning problem
  - Machine Learning and Natural Language Processing.
  - Training Data: some web news and their comments; news documents have been originally annotated keyphrases; obtaining accurate ***Num*$_{keyphrase}$** for each comment.
  - Features: the number of words or sentences; percentage of nouns; similarity; helpfulness……



Experiment

- Dataset: 60 web news from AOL.

	Ave	Max	Min
#Tokens in Article	570.77	1558	162
#Comments	103.98	316	22
#Gold Standard Keyphrases	7.60	17	3

- Evaluation Metric:
  - precision p = $count_{correct}/count_{extrator}$
  - recall r = $count_{correct}/count_{gold}$
  - f1 = 2pr/(p + r)
- Keyphrase Extractors: Tf-idf and SingleRank (Hasan, Coing'10).

- Performance of All Comments Testing (Extracting 20 Keyphrases).

	Precision	Recall	F1
Base_Tf-idf	0.139	0.366	0.202
All_Tf-idf	0.153	0.259	0.192
Base_SingleRank	0.120	0.316	0.174
All_SingleRank	0.096	0.163	0.121

Simply using all comments can not help keyphrase extraction!

- Performance of Useful Comments Testing (Extracting 20 Keyphrases).

	Precision	Recall	F1
Base_Tf-idf	0.139	0.366	0.202
Oracle Tf-idf	0.165	0.432	0.239
Base_SingleRank	0.120	0.316	0.174
Oracle_SingleRank	0.146	0.382	0.211

- Using the gold keyphrases to obtain the accurate $Num_{keyphrase}$ value for each comment to do **Oracle test**.
- Selecting top 15 high score k comments.

There is a subset of useful comments which can help keyphrase extraction!

- Performance of Extracting 20 Keyphrases.

	Precision	Recall	F1
Base_Tf-idf	0.139	0.366	0.202
Similar_Tf-idf	0.151	0.393	0.218
Helpfulness_Tf-idf	0.144	0.379	0.209
KeyInfo_Tf-idf	0.157	0.406	0.226
Base_SingleRank	0.120	0.316	0.174
Similar_ SingleRank	0.134	0.324	0.190
Helpfulness_ SingleRank	0.133	0.332	0.190
KeyInfo_ SingleRank	0.144	0.331	0.201

- 5 new AOL web news and 524 comments, which have gold keyphrase, construct keyinfo model to predict the $Num_{keyphrase}$.
- Selecting top 15 high score comments.
- All comments selectors are effective and KeyInfo comments selector is the best!

- Useful Comments Sets Comparing Evaluation.

	Number
$Set_{Oracle} \cap Set_{KeyInfo}$	437
$Set_{Oracle} \cap Set_{Helpfulness}$	331
$Set_{Oracle} \cap Set_{Similar}$	300

- Compare three selected comment sets with the comments selected based on the accurate $\mathbf{Num_{keyphrase}}$ value.
- KeyInfo selector collects the largest most useful comments!

Conclusion

- We propose using comments information for keyphrase extraction from web news documents.
- We give three strategies to select useful comments.
- All comments strategies are effective and our machine learning approach is the best.

Thanks!