NTCIR-7 Patent Mining Experiments at RALI

Guihong Cao, Jian-Yun Nie and Lixin Shi
Department of Computer Science and Operations Research
University of Montreal, Canada

Outline

- Introduction
- Our Approaches
- Issues Investigated
- Experiments
- Conclusion

Introduction

- Patent Mining Project
  - Each patent has an IPC code
- Task
  - Query: abstract of a research paper
  - Document collection: patents with IPC code
  - Task: assign IPC codes to each research paper according to the relevance
- Possible solution
  - View it as a text categorization problem

Introduction (Cont.)

- Difference in writing style for patent and research paper
  - Patent: more general terms to cover more related things
  - Research paper: more precise and technical

  Eg. Music player VS Apple iPod
- Complexity in classification problem
  - More than 50,000 IPC codes
  - Very unbalanced
  - Cannot be tackled with traditional text classification approaches

Distribution of IPC codes in US patents

#Patent	#IPC	#Patent	#IPC
1~10	25944	2001~3000	5
11~100	10911	3001~4000	3
101~500	1430	4001~5000	0
501~1000	129	>5000	23
1001~2000	46		

Outline

- Introduction
- Our Approaches
  - Basic approach
  - System Description
- The Issues Investigated
- Experiments
- Conclusion

Basic Approach

- Classify the research paper with K-NN classifier
  - The patents are labeled instances
  - Measure the distance between patents and research paper according to relevance
- Finding closest documents with information retrieval
  - Language modeling approach for information retrieval
  - Measuring relevance by query likelihood

Language Modeling Approach for Information Retrieval

- Documents are represented with unigram models, i.e., *P(w|D)*
  - *P(w|D)* is smoothed to avoid zero probablity (Zhai and Lafferty, 2001)

$$P(w \mid D) = \lambda \frac{tf(w, D)}{|D|} + (1 - \lambda) P(w \mid C)$$

- A query is represented as a sequence of words
- Relevance is measured by the likelihood of query with respect to the document model

$$P(q \mid D) = \prod_{q_i} P(q_i \mid D)$$

System Description

- The whole system is implemented using INDRI system (Strohman et al, 2005)
- INDRI system
  - Language modeling approach for IR
  - Allowing retrieval using different fields
- Classification algorithm

$$score(c, q) = \sum_{i=1}^{K} \delta(ipc(d_i) = c) P(q \mid d_i)$$

$\delta(ipc(d_i) = c)$: indicator function

Outline

- Introduction
- Our Approaches
- The Issues Investigated
- Experiments
- Conclusion

Investigations

- Term Distillation
  - Aiming to solve different styles between research paper and patent description
- Some common words in research paper are not common words in patent description

  e.g. paper, study, propose
- Introducing noises to patent retrieval
- Out approach
  - Selected a set of common terms in research paper according to document frequency
  - Filtering out the common words in query time

Common terms

lt	propose	prepare	shows
gt	proposed	prepares	showing
paper	based	preparing	shown
papers	obtain	prepared	report
method	obtains	carry	reported
methods	obtained	carries	
study	find	carrying	
studies	found	carried	
studying	result	show	
studied	results	showed	

Mining Patent Structures

- Patent: structured documents
- Different fields have different impacts
- Four main fields
  - Title, abstract, specification and claim
- Specification can be divided into fours sub-fields
  - Background, description, summary and drawing
- Experiments:
  - Using some of the fields
  - Aggregating occurrence of query terms in different fields with linear interpolation
    - With equal weights

Query Expansion

- An effective technique to enrich query with terms from top-ranked documents
- Pseudo-relevance feedback
- Number of feedback documents and query terms is a key issue
- More effective for short queries
- Is it effective for the Patent Mining task (quite long query)?

Outline

- Introduction
- Our Approaches
- The Issues Investigated
- Experiments
- Conclusion

Experiments

- Query and document processing: in standard way
  - Porter stemmer
  - Removing stop words
- Evaluation metrics
  - Mean average precision
  - Precision at top N documents ($P@N$)

Term Distillation Results

Model	**P@30**	**P@100**	**MAP**
Original	0.0277	0.0047	0.1502
Term Distillation	0.0282	0.0046	0.1491

Does not seem to be effective.

Is it due to the terms selected?

The Effectiveness of Query Expansion

Top 20 documents

#Exp. Terms	P@30	P@100	MAP
0	0.0271	0.0047	0.1488
20	0.0274	0.0029	0.1470
40	0.0274	0.0030	0.1451
60	0.0277	0.0029	0.1447
80	0.0277	0.0030	0.1439
100	0.0276	0.0030	0.1456

Observation: Not very effective.

Possibly due to the fact that queries (paper abstracts) are already quite long.

The Impact of Different Fields

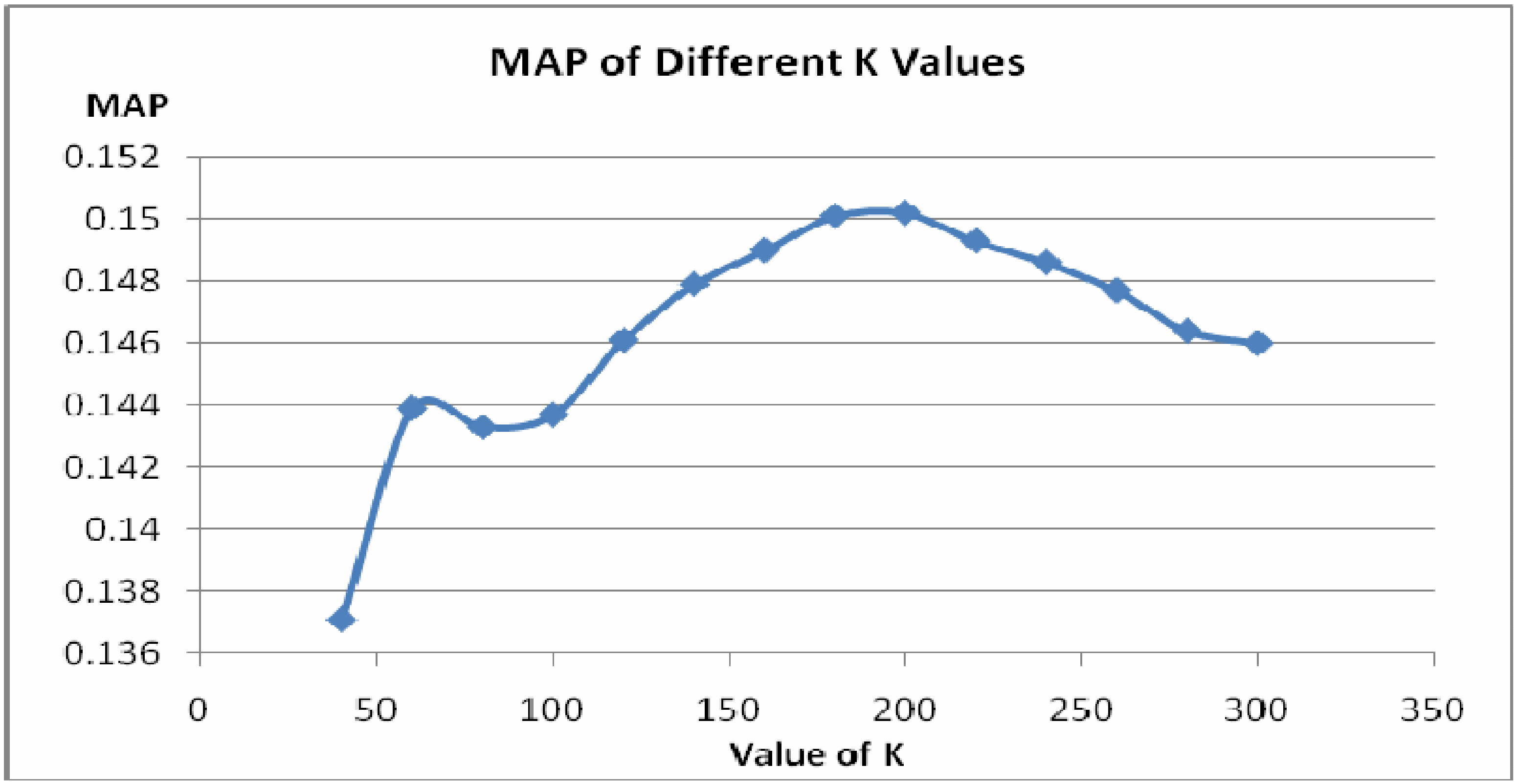
T: title A: abstract S: specification C: claim

B: background D: description M: summary R: drawing

Fields	P@30	P@100	MAP
T+A+S+C	0.0277	0.0047	0.1502
T+A+B	0.0270	0.0041	0.1470
T+A+B+D	0.0281	0.0049	0.1489
T+A+B+D+M	0.0276	0.0047	0.1495

No significant differences

The Impact of Different K Values



Formal Run Results

rali_baseline: Title+Abstract+Specification+Claim

Rali_short_doc: Title+Abstract+Description

Run ID	P@30	P@100	MAP
rali_baseline	0.0234	0.0050	0.1423
rali_short_doc	0.0241	0.0048	0.1437

Marginal effect.

Need to carry out more experiments using different fields.

Conclusion

- Classification of research abstracts into IPC
  - *K-NN* classifier
- Investigated several issues
  - Only the value of K has some impact on classification effectiveness
  - The other factors do not seem to affect the classification accuracy:
    - Different fields
    - pseudo-relevance feedback
    - Term distillation
- Questions:
  - Exploiting more characteristics of patents?
  - Term relationships?

Thanks!