Using supervised machine learning to automatically build relevance judgments for a test collection

Mireille Makary / Dr. Michael Oakes / Prof. Ruslan Mitkov

– University of Wolverhampton, UK

Dr. Fadi Yamout

– Lebanese International University, Lebanon

TIR 2017

August 29, 2017

Outline

- Introduction
- Related Work
- Using machine learning to build qrels
- Conclusion and future work

Introduction

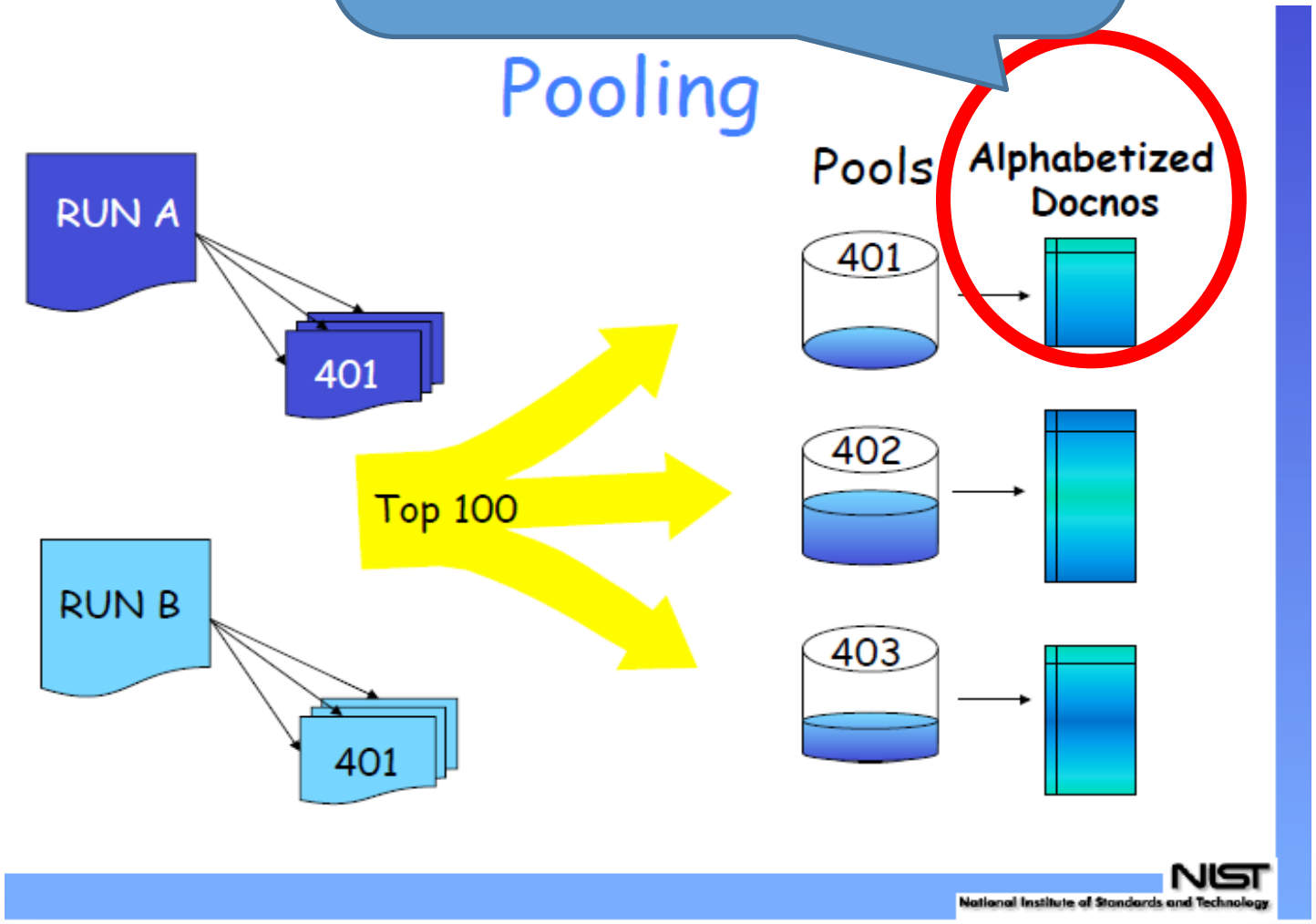
- Test collections constitute the standard framework in information retrieval to evaluate and compare the performance of different information retrieval systems.
- A test collection consists of
  - a set of documents
  - a set of topics (**TREC has 50 topics, CLEF 2003 has 60**)
  - a set of relevance judgments or qrels (query based relevance sets) which indicates the binary relevance of a document to a certain topic.
    - When a “0” value is assigned, the document is judged non-relevant, the value “1” indicates a relevant document.

Introduction

- Building the test collections heavily relies on human assessors to determine the relevance of the documents retrieved for any a topic submitted to the system.
- Because in a large scale environment, such as the web, judging each document is practically infeasible, so only a subset of the documents is judged and used in experiments → TREC (Text Retrieval Conference) forms a **pool of documents**.
  - It is still a large number and consumes time and effort

TREC Poolin

Give to human assessors to determine if each document in this pool is relevant (1) or non-relevant (0)



Related Work

- There have been several studies over the years to propose techniques that reduce the human effort put in building the relevance assessments and some approaches were partially successful in building test collections without human intervention.

Related Work

- Soboroff et al. [1]: first to work on ranking different TREC systems without relevance assessment but rather by random sampling from the pool
- Wu and Crestani [2]:Methods rely on documents' score based on reference count, the number of times a document was retrieved for a query
- Aslam and Savell [3]: ranking systems without relevance assessment
  - Method based on how system runs resemble one another
- Nuray and Can [4]: ranking systems without relevance assessment based on a Data fusion techniques: the top b documents from each of the k systems were combined.

Related Work

- Spoerri[5]: ranked teams rather than runs, each team has one representative run
- Sakai et al [6]: sorted the pooled documents by number of runs that returned the documents, then by the sum of the ranks of that document within the runs.
- Shi et al. [9] used clustering to improve retrieval evaluation without relevance judgments in order to reduce the negative effect of similar runs.

Conclusion from previous studies

- Some of the previously described studies still require human intervention to build the qrels
- Other studies can work for only particular type of test collections or they require some knowledge about the test collection
- Some techniques require a lot of effort
- No perfect correlations

Using machine learning to build qrels

- There has not been so far any work that uses such techniques to build the relevance assessments.
- We have conducted experiments using the supervised machine learning using Naïve Bayes classifier and the Support Vector Machine (we suggest two different approaches) – using TFIDF and Doc2Vec representations

Using supervised machine learning to build qrels

First approach using TFIDF:

1. For each topic, use the documents retrieved by S% of the systems and use them as the training set for relevant documents
2. Use the same number of documents but retrieved from the bottom results and use them as the training set for non-relevant documents
3. Classify the remaining documents as either relevant or non relevant

First Approach Results

To evaluate the qrels, we compute the MAP score for each system using the human-built qrels and the newly generated qrels, then we rank these systems.

	Using SVM		Using NB (alpha=1 default)	
	Kendall's tau	Spearman	Kendall's tau	Spearman
TREC6	0.5266	0.7145	**0.5408**	**0.7322**
TREC7	**0.4328**	**0.5116**	0.4250	0.4769
TREC8	**0.5259**	**0.7646**	0.4617	0.7312

TREC6 has 74 systems
TREC7 has 103 systems
TREC8 has 129 systems

First Approach Results

	Using NB – alpha 0.1	
	Kendall's tau	**Spearman**
TREC6	0.4669	0.6433
TREC7	0.4413	0.4985
TREC8	0.4559	0.7320

Using supervised machine learning to build qrels

Second approach using TFIDF:

1. For each topic, use the documents retrieved by S% of the systems and use them as the training set for known topic-classification
2. Run the classifier to label all the remaining documents in the pool
3. The documents labelled by the classifier for a topic are considered relevant while the remaining documents retrieved by the systems will be judged as non-relevant.

Second Approach Results

	Using SVM		Using NB (alpha=1 default)	
	Kendall's tau	Spearman	Kendall's tau	Spearman
TREC6	0.5712	0.7631	**0.5864**	**0.7749**
TREC7	0.4116	0.5223	**0.5128**	**0.6386**
TREC8	0.4494	0.7266	**0.5144**	**0.7821**

Second Approach Results

Best Results

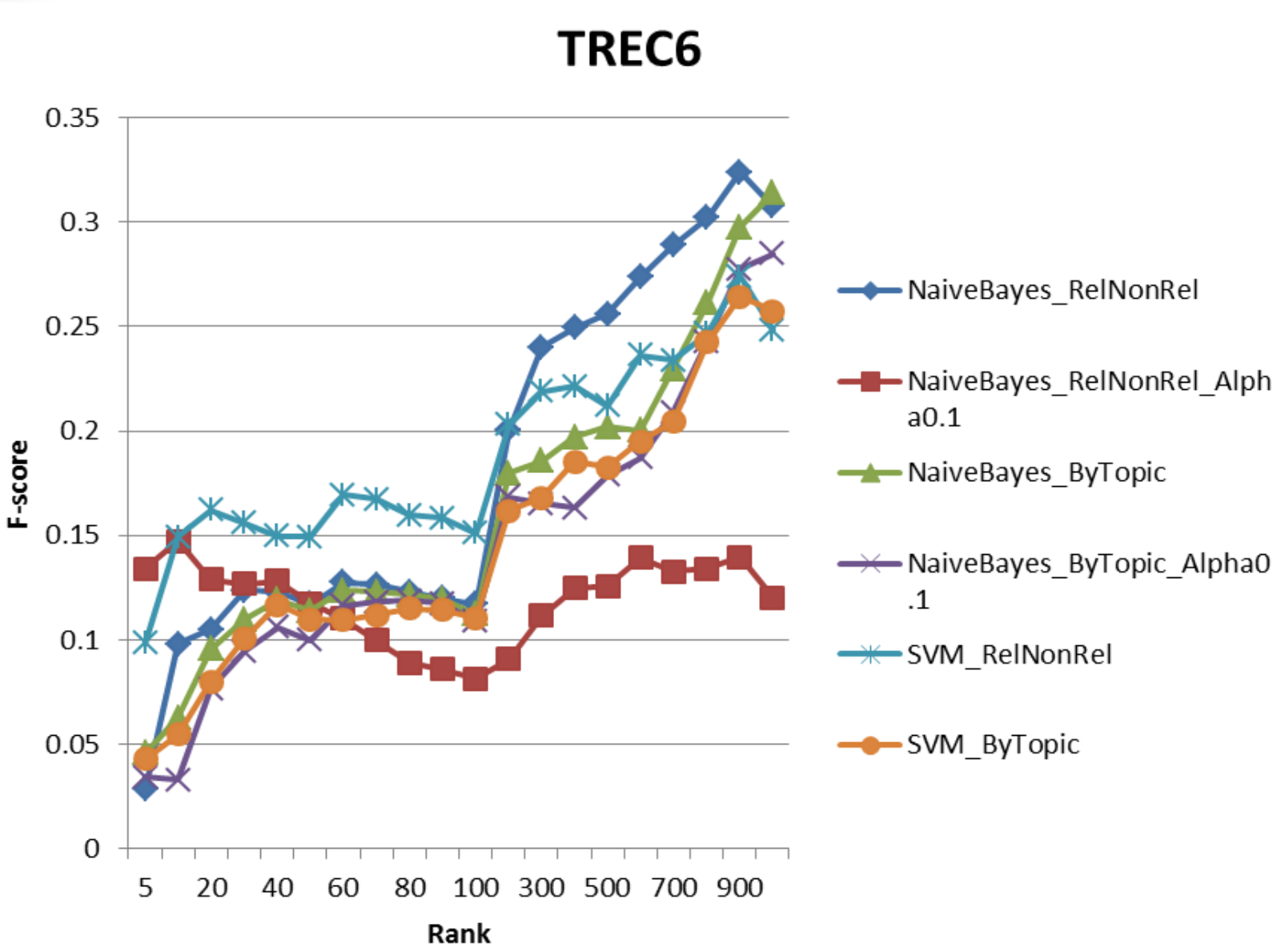
	Using NB – alpha 0.1	
	Kendall's tau	**Spearman**
TREC6	0.5887	0.7787
TREC7	0.5661	0.6746
TREC8	0.5330	0.7907

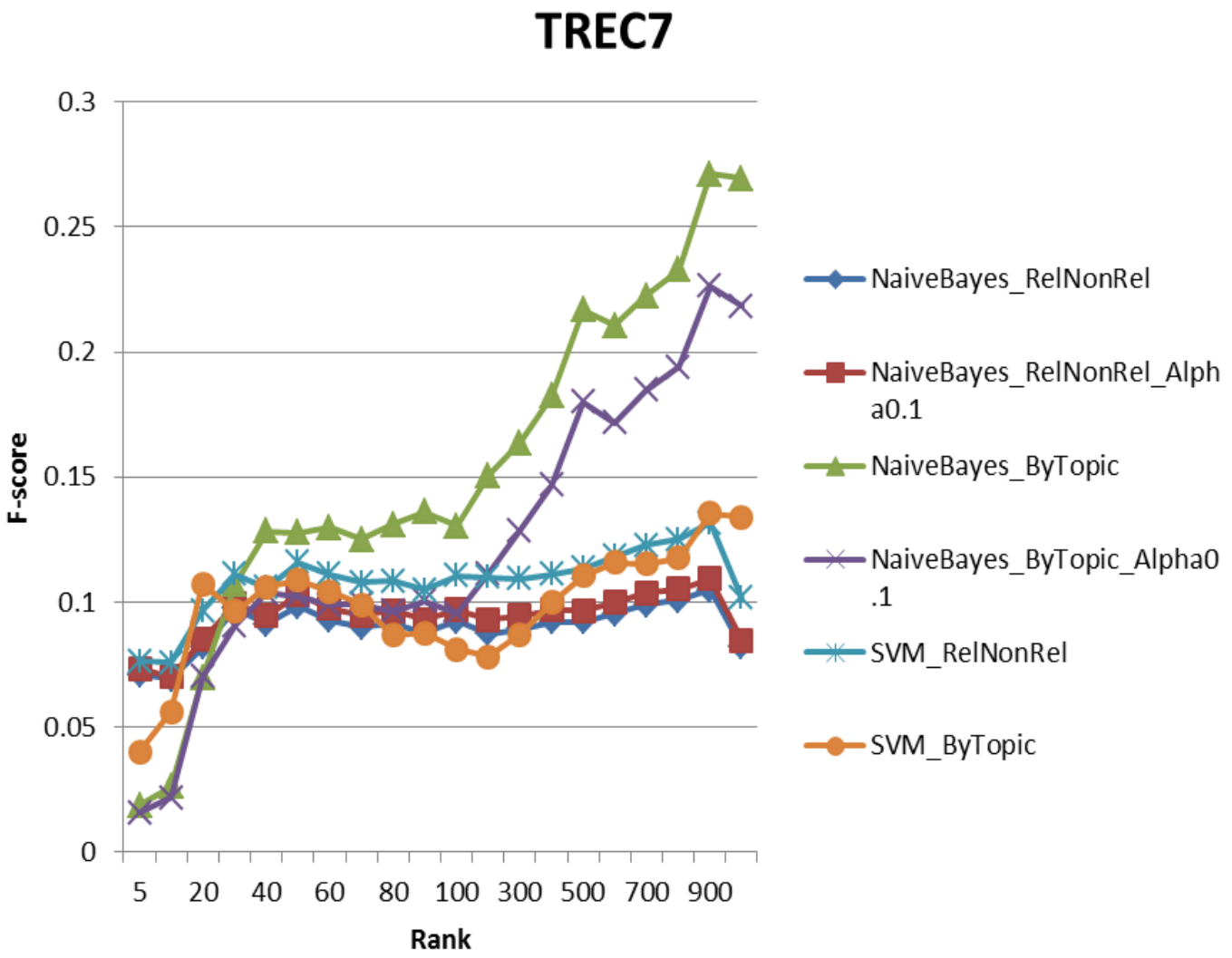
A comparison of Spearman values with previous baseline methods

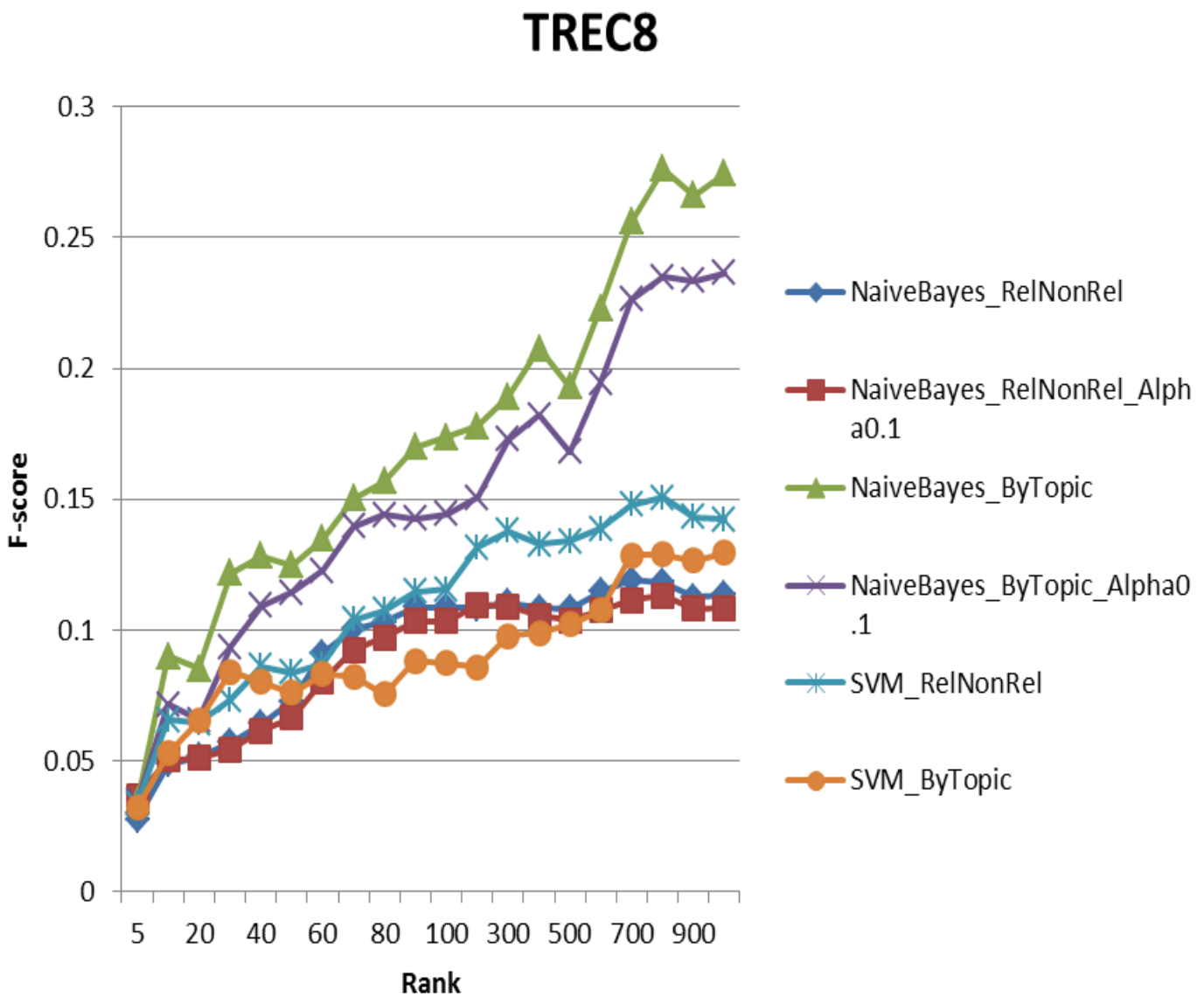
	RS	RC	CB	Single %	ASS	ASSBC	NB By Topic
TREC6	0.436	0.384	0.717	0.618	0.630	**0.854**	**0.778**
TREC7	0.411	0.382	0.453	0.550	0.585	0.631	**0.674**

An intrinsic evaluation for qrels

- We computed the precision and recall measures at different ranks (@5, @10, and @20…@100, @200...@1000).
- The formula used for the precision metric is shown in equation below:
  - **Precision = dAH / dA**
    - Where dAH is the total number of documents judged relevant by both the classifier and the human judges, and dA is the number of documents judged relevant by the classifier.
- As for the recall metric, the formula used is shown below:
  - **Recall = dAH / dH**
    - dH is the total number of documents judged relevant by human assessors
- F-score is the harmonic mean:
  - **F=2 / ((1/p+1/r))**


TREC6
F-score
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
5
20
40
60
80
100
300
500
700
900
Rank
NaiveBayes_RelNonRel
NaiveBayes_RelNonRel_Alpha0.1
NaiveBayes_ByTopic
NaiveBayes_ByTopic_Alpha0.1
SVM_RelNonRel
SVM_ByTopic


TREC7
F-score
0.3
0.25
0.2
0.15
0.1
0.05
0
5
20
40
60
80
100
300
500
700
900
Rank
NaiveBayes_RelNonRel
NaiveBayes_RelNonRel_Alpha0.1
NaiveBayes_ByTopic
NaiveBayes_ByTopic_Alpha0.1
SVM_RelNonRel
SVM_ByTopic


TREC8
F-score
0.3
0.25
0.2
0.15
0.1
0.05
0
5
20
40
60
80
100
300
500
700
900
Rank
NaiveBayes_RelNonRel
NaiveBayes_RelNonRel_Alpha0.1
NaiveBayes_ByTopic
NaiveBayes_ByTopic_Alpha0.1
SVM_RelNonRel
SVM_ByTopic

Using doc2vec to represent documents

- The same experiments were repeated but using the doc2vec representation of documents rather than TFIDF.
- We divided our data into three categories:
  - training data,
  - cross validation data
  - and test data.

Using doc2vec to represent documents

- Training data 50% of the documents retrieved by the S% of the systems
- The other 50% was used for cross validation of the doc2vec model.
- The remaining documents in the pool constituted the test data which had to be labeled using the trained doc2vec model and both the NB and SVM classifiers.
- **The test data in our case is much larger than the training data.**

Using doc2vec to represent documents

- The parameters used for the do following:

min_count=1, window=10 =1
negative=5, workers=8

TFIDF Better:

Spearman 0.7787

Spear an based on MAP scores				
	Using SVM		Using NB	
	By Topic	Relevant/Non-Relevant	By Topic	Relevant/Non-Relevant
TREC6	0.6257	0.6813	0.7555	0.7550
TREC7	0.6175	0.5594	0.6116	0.5158
TREC8	0.7293	0.6334	0.7081	0.6839

Using doc2vec to represent documents

- The parameters used for the doc2vec m
  following:

min_count=1, window=10, size=100, s
negative=5, workers=8

TFIDF Better:
Spearman
0.6746

Spearman based MAP scores				
	Using SV		Using NB	
	By Topic	Relevant/Non-Relevant	By Topic	Relevant/Non-Relevant
TREC6	0.6257	0.6813	0.7555	0.7550
TREC7	0.6175	0.5594	0.6116	0.5158
TREC8	0.7293	0.6334	0.7081	0.6839

Using doc2vec to represent documents

- The parameters used for the doc2vec m
  following:

min_count=1, window=10, size=100, sa
negative=5, workers=8

TFIDF Better:
Spearman
0.7907

Spearman based on M				
	Using SVM		Using NB	
	By Topic	Relev Non-elevant	By Topic	Relevant/Non-Relevant
TREC6	0.6257	0.6813	0.7555	0.7550
TREC7	0.6175	0.5594	0.6116	0.5158
TREC8	0.7293	0.6334	0.7081	0.6839

Conclusion and Future Work

- We were able to devise new techniques using machine learning to build the qrels automatically without any human intervention.
- We are now testing the proposed methodologies related to machine learning on foreign languages: French, Finnish, etc. (CLEF2002-2003)
- Further analysis of the quality of the qrels, evaluate the discrimination between the best systems, average and poor ones.

References

[1] Soboroff I., Nicholas C., and Cahan P. Ranking retrieval systems without relevance judgments, In Proceedings of ACM SIGIR 2001, pages 66–73, 2001.

[2] Aslam J. A. and Savell R. On the effectiveness of evaluating retrieval systems in the absence of relevance judgments. In Proceedings of ACM SIGIR 2003, pages 361–362, 2003.

[3]Wu S. and Crestani F. Methods for ranking information retrieval systems without relevance judgments. In Proceedings of the 2003 ACM Symposium on Applied Computing, pages 811-816, 2003.

[4]Nuray R. and Can F. Automatic ranking of information retrieval systems using data fusion, Information Processing and Management, 42:595–614, 2006.

[5]Spoerri A. Using the structure of overlap between search results to rank retrieval systems without relevance judgments. Information Processing and Management, 43:1059–1070, 2007.

[7]Efron M.: Using multiple query aspects to build test collections without human relevance judgements, ECIR, 2009

[8] Rajagopal P., Ravana S.D., and Ismail M.A. Relevance judgments exclusive of human as-sessors in large scale information retrieval evaluation experimentation, 2014.

[9] Shi Z., Li P., Wang B. Using Clustering to Improve Retrieval Evaluation without Relevance Judgments, Coling 2010: Poster Volume, pages 1131–1139, Beijing, August 2010

Thank you!