SemEval-2010 Task 5: Automatic Keyphrase Extraction from Scientific Articles

Su Nam Kim,♠ Olena Medelyan,♡ Min-Yen Kan♢ and **Timothy Baldwin♠**
♠ Dept of Computer Science and Software Engineering, University of Melbourne, Australia
♡ Pingar LP, Auckland, New Zealand
♢ School of Computing, National University of Singapore, Singapore
sunamkim@gmail.com, medelyan@gmail.com,
kanmy@comp.nus.edu.sg, tb@ldwin.net

Abstract

This paper describes Task 5 of the Workshop on Semantic Evaluation 2010 (SemEval-2010). Systems are to automatically assign keyphrases or keywords to given scientific articles. The participating systems were evaluated by matching their extracted keyphrases against manually assigned ones. We present the overall ranking of the submitted systems and discuss our findings to suggest future directions for this task.

1 Task Description

Keyphrases[1] are words that capture the main topics of a document. As they represent these key ideas, extracting high-quality keyphrases can benefit various natural language processing (NLP) applications such as summarization, information retrieval and question-answering. In summarization, keyphrases can be used as a form of semantic metadata (Barzilay and Elhadad, 1997; Lawrie et al., 2001; D'Avanzo and Magnini, 2005). In search engines, keyphrases can supplement full-text indexing and assist users in formulating queries.

Recently, a resurgence of interest in keyphrase extraction has led to the development of several new systems and techniques for the task (Frank et al., 1999; Witten et al., 1999; Turney, 1999; Hulth, 2003; Turney, 2003; Park et al., 2004; Barker and Cornnacchia, 2000; Hulth, 2004; Matsuo and Ishizuka, 2004; Mihalcea and Tarau, 2004; Medelyan and Witten, 2006; Nguyen and Kan, 2007; Wan and Xiao, 2008; Liu et al., 2009; Medelyan, 2009; Nguyen and Phan, 2009). These have showcased the potential benefits of keyphrase extraction to downstream NLP applications.

In light of these developments, we felt that this was an appropriate time to conduct a shared task for keyphrase extraction, to provide a standard assessment to benchmark current approaches. A second goal of the task was to contribute an additional public dataset to spur future research in the area.

Currently, there are several publicly available data sets.[2] For example, Hulth (2003) contributed 2,000 abstracts of journal articles present in Inspec between the years 1998 and 2002. The data set contains keyphrases (i.e. controlled and uncontrolled terms) assigned by professional indexers — 1,000 for training, 500 for validation and 500 for testing. Nguyen and Kan (2007) collected a dataset containing 120 computer science articles, ranging in length from 4 to 12 pages. The articles contain author-assigned keyphrases as well as reader-assigned keyphrases contributed by undergraduate CS students. In the general newswire domain, Wan and Xiao (2008) developed a dataset of 308 documents taken from DUC 2001 which contain up to 10 manually-assigned keyphrases per document. Several databases, including the ACM Digital Library, IEEE Xplore, Inspec and PubMed provide articles with author-assigned keyphrases and, occasionally, reader-assigned ones. Medelyan (2009) automatically generated a dataset using tags assigned by the users of the collaborative citation platform CiteULike. This dataset additionally records how many people have assigned the same keyword to the same publication. In total, 180 full-text publications were annotated by over 300 users.[3] Despite the availability of these datasets, a standardized benchmark dataset with a well-defined train-

[1] We use "keyphrase" and "keywords" interchangeably to refer to both single words and phrases.

♢ Min-Yen Kan's work was funded by National Research Foundation grant "Interactive Media Search" (grant # R-252-000-325-279).

[2] All data sets listed below are available for download from http://github.com/snkim/AutomaticKeyphraseExtraction

[3] http://bit.ly/maui-datasets

Proceedings of the 5th International Workshop on Semantic Evaluation, ACL 2010, pages 21–26,
Uppsala, Sweden, 15-16 July 2010.

ing and test split is needed to maximize comparability of results.

For the SemEval-2010 Task 5, we have compiled a set of 284 scientific articles with keyphrases carefully chosen by both their authors and readers. The participants' task was to develop systems which automatically produce keyphrases for each paper. Each team was allowed to submit up to three system runs, to benchmark the contributions of different parameter settings and approaches. Each run consisted of extracting a ranked list of 15 keyphrases from each document, ranked by their probability of being reader-assigned keyphrases.

In the remainder of the paper, we describe the competition setup, including how data collection was managed and the evaluation methodology (Section 2). We present the results of the shared task, and discuss the immediate findings of the competition in Section 3. In Section 4 we assess the human performance by comparing reader-assigned keyphrases to those assigned by the authors. This gives an approximation of an upper-bound performance for this task.

2 Competition Setup

2.1 Data

We collected trial, training and test data from the ACM Digital Library (conference and workshop papers). The input papers ranged from 6 to 8 pages, including tables and pictures. To ensure a variety of different topics was represented in the corpus, we purposefully selected papers from four different research areas for the dataset. In particular, the selected articles belong to the following four 1998 ACM classifications: C2.4 (Distributed Systems), H3.3 (Information Search and Retrieval), I2.11 (Distributed Artificial Intelligence – Multiagent Systems) and J4 (Social and Behavioral Sciences – Economics). All three datasets (trial, training and test) had an equal distribution of documents from among the categories (see Table 1). This domain specific information was provided with the papers (e.g. I2.4-1 or H3.3-2), in case participant systems wanted to utilize this information. We specifically decided to straddle different areas to see whether participant approaches would work better within specific areas.

Participants were provided with 40, 144, and 100 articles, respectively, in the trial, training and test data, distributed evenly across the four research areas in each case. Note that the trial data is a subset of the training data. Since the original format for the articles was PDF, we converted them into (UTF-8) plain text using pdftotext, and systematically restored full words that were originally hyphenated and broken across two lines. This policy potentially resulted in valid hyphenated forms having their hyphen (-) removed.

All collected papers contain author-assigned keyphrases, part of the original PDF file. We additionally collected reader-assigned keyphrases for each paper. We first performed a pilot annotation task with a group of students to check the stability of the annotations, finalize the guidelines, and discover and resolve potential issues that may occur during the actual annotation. To collect the actual reader-assigned keyphrases, we then hired 50 student annotators from the Computer Science department of the National University of Singapore.

We assigned 5 papers to each annotator, estimating that assigning keyphrases to each paper should take about 10-15 minutes. Annotators were explicitly told to extract keyphrases that actually appear in the text of each paper, rather than to create semantically-equivalent phrases, but could extract phrases from any part of the document (including headers and captions). In reality, on average 15% of the reader-assigned keyphrases did not appear in the text of the paper, but this is still less than the 19% of author-assigned keyphrases that did not appear in the papers. These values were computed using the test documents only. In other words, the maximum recall that the participating systems can achieve on these documents is 85% and 81% for the reader- and author-assigned keyphrases, respectively.

As some keyphrases may occur in multiple forms, in our evaluation we accepted two different versions of genitive keyphrases: A *of* $B \rightarrow B$ A (e.g. *policy of school* = *school policy*) and A*'s* $B \rightarrow A$ B (e.g. *school's policy* = *school policy*). In certain cases, such alternations change the semantics of the candidate phrase (e.g., *matter of fact* vs. *?fact matter*). We judged borderline cases by committee and do not include alternations that were judged to be semantically distinct.

Table 1 shows the distribution of the trial, training and test documents over the four different research areas, while Table 2 shows the distribution of author- and reader-assigned keyphrases.

Interestingly, among the 387 author-assigned

Dataset	Total	Document Topic			
		C	H	I	J
Trial	40	10	10	10	10
Training	144	34	39	35	36
Test	100	25	25	25	25

Table 1: Number of documents per topic in the trial, training and test datasets, across the four ACM document classifications

Dataset	Author	Reader	Combined
Trial	149	526	621
Training	559	1824	2223
Test	387	1217	1482

Table 2: Number of author- and reader-assigned keyphrases in the different datasets

keywords, 125 keywords match exactly with reader-assigned keywords, while many more near-misses (i.e. partial matches) occur.

2.2 Evaluation Method and Baseline

Traditionally, automatic keyphrase extraction systems have been assessed using the proportion of top-N candidates that exactly match the gold-standard keyphrases (Frank et al., 1999; Witten et al., 1999; Turney, 1999). In some cases, inexact matches, or near-misses, have also been considered. Some have suggested treating semantically-similar keyphrases as correct based on similarities computed over a large corpus (Jarmasz and Barriere, 2004; Mihalcea and Tarau, 2004), or using semantic relations defined in a thesaurus (Medelyan and Witten, 2006). Zesch and Gurevych (2009) compute near-misses using an n-gram based approach relative to the gold standard. For our shared task, we follow the traditional exact match evaluation metric. That is, we match the keyphrases in the answer set with those the systems provide, and calculate micro-averaged precision, recall and F-score ($\beta = 1$). In the evaluation, we check the performance over the top 5, 10 and 15 candidates returned by each system. We rank the participating systems by F-score over the top 15 candidates.

Participants were required to extract existing phrases from the documents. Since it is theoretically possible to retrieve author-assigned keyphrases from the original PDF articles, we evaluate the participating systems over the independently-generated and held-out reader-assigned keyphrases, as well as the combined set of keyphrases (author- and reader-assigned).

All keyphrases in the answer set are stemmed using the English Porter stemmer for both the training and test dataset.[4]

We computed a TF×IDF n-gram based baseline using both supervised and unsupervised learning systems. We use 1, 2, 3-grams as keyphrase candidates, used Naïve Bayes (NB) and Maximum Entropy (ME) classifiers to learn two supervised baseline systems based on the keyphrase candidates and gold-standard annotations for the training documents. In total, there are three baselines: two supervised and one unsupervised. The performance of the baselines is presented in Table 3, where *R* indicates reader-assigned keyphrases and *C* indicates combined (both author- and reader-assigned) keyphrases.

3 Competition Results

The trial data was downloaded by 73 different teams, of which 36 teams subsequently downloaded the training and test data. 21 teams participated in the final competition, of which two teams withdrew their systems.

Table 4 shows the performance of the final 19 submitted systems. 5 teams submitted one run, 6 teams submitted two runs and 8 teams submitted the maximum number of three runs. We rank the best-performing system from each team by micro-averaged F-score over the top 15 candidates. We also show system performance over reader-assigned keywords in Table 5, and over author-assigned keywords in Table 6. In all these tables, P, R and F denote precision, recall and F-score, respectively.

The best results over the reader-assigned and combined keyphrase sets are **23.5%** and **27.5%**, respectively, achieved by the *HUMB* team. Most systems outperformed the baselines. Systems also generally did better over the combined set, as the presence of a larger gold-standard answer set improved recall.

In Tables 7 and 8, we ranked the teams by F-score, computed over the top 15 candidates for each of the four ACM document classifications. The numbers in brackets are the actual F-scores

[4]Using the Perl implementation available at `http://tartarus.org/~martin/PorterStemmer/`; we informed participants that this was the stemmer we would be using for the task, to avoid possible stemming variations between implementations.

Method	by	Top 5 candidates			Top 10 candidates			Top 15 candidates		
		P	R	F	P	R	F	P	R	F
TF×IDF	R	17.8%	7.4%	10.4%	13.9%	11.5%	12.6%	11.6%	14.5%	12.9%
	C	22.0%	7.5%	11.2%	17.7%	12.1%	14.4%	14.9%	15.3%	15.1%
NB	R	16.8%	7.0%	9.9%	13.3%	11.1%	12.1%	11.4%	14.2%	12.7%
	C	21.4%	7.3%	10.9%	17.3%	11.8%	14.0%	14.5%	14.9%	14.7%
ME	R	16.8%	7.0%	9.9%	13.3%	11.1%	12.1%	11.4%	14.2%	12.7%
	C	21.4%	7.3%	10.9%	17.3%	11.8%	14.0%	14.5%	14.9%	14.7%

Table 3: Baseline keyphrase extraction performance for one unsupervised (TF×IDF) and two supervised (NB and ME) systems

System	Rank	Top 5 candidates			Top 10 candidates			Top 15 candidates		
		P	R	F	P	R	F	P	R	F
HUMB	1	39.0%	13.3%	19.8%	32.0%	21.8%	26.0%	27.2%	27.8%	27.5%
WINGNUS	2	40.2%	13.7%	20.5%	30.5%	20.8%	24.7%	24.9%	25.5%	25.2%
KP-Miner	3	36.0%	12.3%	18.3%	28.6%	19.5%	23.2%	24.9%	25.5%	25.2%
SZTERGAK	4	34.2%	11.7%	17.4%	28.5%	19.4%	23.1%	24.8%	25.4%	25.1%
ICL	5	34.4%	11.7%	17.5%	29.2%	19.9%	23.7%	24.6%	25.2%	24.9%
SEERLAB	6	39.0%	13.3%	19.8%	29.7%	20.3%	24.1%	24.1%	24.6%	24.3%
KX_FBK	7	34.2%	11.7%	17.4%	27.0%	18.4%	21.9%	23.6%	24.2%	23.9%
DERIUNLP	8	27.4%	9.4%	13.9%	23.0%	15.7%	18.7%	22.0%	22.5%	22.3%
Maui	9	35.0%	11.9%	17.8%	25.2%	17.2%	20.4%	20.3%	20.8%	20.6%
DFKI	10	29.2%	10.0%	14.9%	23.3%	15.9%	18.9%	20.3%	20.7%	20.5%
BUAP	11	13.6%	4.6%	6.9%	17.6%	12.0%	14.3%	19.0%	19.4%	19.2%
SJTULTLAB	12	30.2%	10.3%	15.4%	22.7%	15.5%	18.4%	18.4%	18.8%	18.6%
UNICE	13	27.4%	9.4%	13.9%	22.4%	15.3%	18.2%	18.3%	18.8%	18.5%
UNPMC	14	18.0%	6.1%	9.2%	19.0%	13.0%	15.4%	18.1%	18.6%	18.3%
JU_CSE	15	28.4%	9.7%	14.5%	21.5%	14.7%	17.4%	17.8%	18.2%	18.0%
LIKEY	16	29.2%	10.0%	14.9%	21.1%	14.4%	17.1%	16.3%	16.7%	16.5%
UvT	17	24.8%	8.5%	12.6%	18.6%	12.7%	15.1%	14.6%	14.9%	14.8%
POLYU	18	15.6%	5.3%	7.9%	14.6%	10.0%	11.8%	13.9%	14.2%	14.0%
UKP	19	9.4%	3.2%	4.8%	5.9%	4.0%	4.8%	5.3%	5.4%	5.3%

Table 4: Performance of the submitted systems over the combined author- and reader-assigned keywords, ranked by F-score

for each team. Note that in the case of a tie in F-score, we ordered teams by descending F-score over all the data.

4 Discussion of the Upper-Bound Performance

The current evaluation is a testament to the gains made by keyphrase extraction systems. The system performance over the different keyword categories (reader-assigned and author-assigned) and numbers of keyword candidates (top 5, 10 and 15 candidates) attest to this fact.

The top-performing systems return F-scores in the upper twenties. Superficially, this number is low, and it is instructive to examine how much room there is for improvement. Keyphrase extraction is a subjective task, and an F-score of 100% is infeasible. On the author-assigned keyphrases in our test collection, the highest a system could theoretically achieve was 81% recall[5] and 100% precision, which gives a maximum F-score of 89%. However, such a high value would only be possible if the number of keyphrases extracted per document could vary; in our task, we fixed the thresholds at 5, 10 and 15 keyphrases.

[5]The remaining 19% of keyphrases do not actually appear in the documents and thus cannot be extracted.

Another way of computing the upper-bound performance would be to look into how well people perform the same task. We analyzed the performance of our readers, taking the author-assigned keyphrases as the gold standard. The authors assigned an average of 4 keyphrases to each paper, whereas the readers assigned 12 on average. These 12 keyphrases cover 77.8% of the authors' keyphrases, which corresponds to a precision of 21.5%. The F-score achieved by the readers on the author-assigned keyphrases is 33.6%, whereas the F-score of the best-performing system on the same data is 19.3% (for top 15, not top 12 keyphrases, see Table 6).

We conclude that there is definitely still room for improvement, and for any future shared tasks, we recommend against fixing any threshold on the number of keyphrases to be extracted per document. Finally, as we use a strict exact matching metric for evaluation, the presented evaluation figures are a lower bound for performance, as semantically equivalent keyphrases are not counted as correct. For future runs of this challenge, we believe a more semantically-motivated evaluation should be employed to give a more accurate impression of keyphrase acceptability.

System	Rank	Top 5 candidates			Top 10 candidates			Top 15 candidates		
		P	R	F	P	R	F	P	R	F
HUMB	1	30.4%	12.6%	17.8%	24.8%	20.6%	22.5%	21.2%	26.4%	23.5%
KX_FBK	2	29.2%	12.1%	17.1%	23.2%	19.3%	21.1%	20.3%	25.3%	22.6%
SZTERGAK	3	28.2%	11.7%	16.6%	23.2%	19.3%	21.1%	19.9%	24.8%	22.1%
WINGNUS	4	30.6%	12.7%	18.0%	23.6%	19.6%	21.4%	19.8%	24.7	22.0%
ICL	5	27.2%	11.3%	16.0%	22.4%	18.6%	20.3%	19.5%	24.3%	21.6%
SEERLAB	6	31.0%	12.9%	18.2%	24.1%	20.0%	21.9%	19.3%	24.1%	21.5%
KP-Miner	7	28.2%	11.7%	16.5%	22.0%	18.3%	20.0%	19.3%	24.1%	21.5%
DERIUNLP	8	22.2%	9.2%	13.0%	18.9%	15.7%	17.2%	17.5%	21.8%	19.5%
DFKI	9	24.4%	10.1%	14.3%	19.8%	16.5%	18.0%	17.4%	21.7%	19.3%
UNICE	10	25.0%	10.4%	14.7%	20.1%	16.7%	18.2%	16.0%	19.9%	17.8%
SJTULTLAB	11	26.6%	11.1%	15.6%	19.4%	16.1%	17.6%	15.6%	19.4%	17.3%
BUAP	12	10.4%	4.3%	6.1%	13.9%	11.5%	12.6%	14.9%	18.6%	16.6%
Maui	13	25.0%	10.4%	14.7%	18.1%	15.0%	16.4%	14.9%	18.5%	16.1%
UNPMC	14	13.8%	5.7%	8.1%	15.1%	12.5%	13.7%	14.5%	18.0%	16.1%
JU_CSE	15	23.4%	9.7%	13.7%	18.1%	15.0%	16.4%	14.4%	17.9%	16.0%
LIKEY	16	24.6%	10.2%	14.4%	17.9%	14.9%	16.2%	13.8%	17.2%	15.3%
POLYU	17	13.6%	5.7%	8.0%	12.6%	10.5%	11.4%	12.0%	14.9%	13.3%
UvT	18	20.4%	8.5%	12.0%	15.6%	13.0%	14.2%	11.9%	14.9%	13.2%
UKP	19	8.2%	3.4%	4.8%	5.3%	4.4%	4.8%	4.7%	5.8%	5.2%

Table 5: Performance of the submitted systems over the reader-assigned keywords, ranked by F-score

System	Rank	Top 5 candidates			Top 10 candidates			Top 15 candidates		
		P	R	F	P	R	F	P	R	F
HUMB	1	21.2%	27.4%	23.9%	15.4%	39.8%	22.2%	12.1%	47.0%	19.3%
KP-Miner	2	19.0%	24.6%	21.4%	13.4%	34.6%	19.3%	10.7%	41.6%	17.1%
ICL	3	17.0%	22.0%	19.2%	13.5%	34.9%	19.5%	10.5%	40.6%	16.6%
Maui	4	20.4%	26.4%	23.0%	13.7%	35.4%	19.8%	10.2%	39.5%	16.2%
SEERLAB	5	18.8%	24.3%	21.2%	13.1%	33.9%	18.9%	10.1%	39.0%	16.0%
SZTERGAK	6	14.6%	18.9%	16.5%	12.2%	31.5%	17.6%	9.9%	38.5%	15.8%
WINGNUS	7	18.6%	24.0%	21.0%	12.6%	32.6%	18.2%	9.3%	36.2%	14.8%
DERIUNLP	8	12.6%	16.3%	14.2%	9.7%	25.1%	14.0%	9.3%	35.9%	14.7%
KX_FBK	9	13.6%	17.6%	15.3%	10.0%	25.8%	14.4%	8.5%	32.8%	13.5%
BUAP	10	5.6%	7.2%	6.3%	8.1%	20.9%	11.7%	8.3%	32.0%	13.2%
JU_CSE	11	12.0%	15.5%	13.5%	8.5%	22.0%	12.3%	7.5%	29.0%	11.9%
UNPMC	12	7.0%	9.0%	7.9%	7.7%	19.9%	11.1%	7.1%	27.4%	11.2%
DFKI	13	12.8%	16.5%	14.4%	8.5%	22.0%	12.3%	6.6%	25.6%	10.5%
SJTULTLAB	14	9.6%	12.4%	10.8%	7.8%	20.2%	11.3%	6.2%	24.0%	9.9%
Likey	15	11.6%	15.0%	13.1%	7.9%	20.4%	11.4%	5.9%	22.7%	9.3%
UvT	16	11.4%	14.7%	12.9%	7.6%	19.6%	11.0%	5.8%	22.5%	9.2%
UNICE	17	8.8%	11.4%	9.9%	6.4%	16.5%	9.2%	5.5%	21.5%	8.8%
POLYU	18	3.8%	4.9%	4.3%	4.1%	10.6%	5.9%	4.1%	16.0%	6.6%
UKP	19	1.6%	2.1%	1.8%	0.9%	2.3%	1.3%	0.8%	3.1%	1.3%

Table 6: Performance of the submitted systems over the author-assigned keywords, ranked by F-score

5 Conclusion

This paper has described Task 5 of the Workshop on Semantic Evaluation 2010 (SemEval-2010), focusing on keyphrase extraction. We outlined the design of the datasets used in the shared task and the evaluation metrics, before presenting the official results for the task and summarising the immediate findings. We also analyzed the upper-bound performance for this task, and demonstrated that there is still room for improvement over the task. We look forward to future advances in automatic keyphrase extraction based on this and other datasets.

References

Ken Barker and Nadia Corrnacchia. Using noun phrase heads to extract document keyphrases. In *Proceedings of BCCSCSI : Advances in Artificial Intelligence*. 2000, pp.96–103.

Regina Barzilay and Michael Elhadad. Using lexical chains for text summarization. In *Proceedings of ACL/EACL Workshop on Intelligent Scalable Text Summarization*. 1997, pp. 10–17.

Ernesto D'Avanzo and Bernado Magnini. A Keyphrase-Based Approach to Summarization: the LAKE System at DUC-2005. In *Proceedings of DUC*. 2005.

Eibe Frank and Gordon W. Paynter and Ian H. Witten and Carl Gutwin and Craig G. Nevill-Manning. Domain Specific Keyphrase Extraction. In *Proceedings of IJCAI*. 1999, pp.668–673.

Annette Hulth. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of EMNLP*. 2003, 216–223.

Annette Hulth. Enhancing Linguistically Oriented Automatic Keyword Extraction. In *Proceedings of HLT/NAACL*. 2004, pp. 17–20.

Mario Jarmasz and Caroline Barriere. Using semantic similarity over tera-byte corpus, compute the performance of keyphrase extraction. In *Proceedings of CLINE*. 2004.

Dawn Lawrie and W. Bruce Croft and Arnold Rosenberg. Finding Topic Words for Hierarchical Summarization. In *Proceedings of SIGIR*. 2001, pp. 349–357.

Rank	Group C	Group H	Group I	Group J
1	HUMB(28.3%)	HUMB(30.2%)	HUMB(24.2%)	HUMB(27.4%)
2	ICL(27.2%)	WINGNUS(28.9%)	SEERLAB(24.2%)	WINGNUS(25.4%)
3	KP-Miner(25.5%)	SEERLAB(27.8%)	KP-Miner(22.8%)	ICL(25.4%)
4	SZTERGAK(25.3%)	KP-Miner(27.6%)	KX_FBK(22.8%)	SZTERGAK(25.17%)
5	WINGNUS(24.2%)	SZTERGAK(27.6%)	WINGNUS(22.3%)	KP-Miner(24.9%)
6	KX_FBK(24.2%)	ICL(25.5%)	SZTERGAK(22.25%)	KX_FBK(24.6%)
7	DERIUNLP(23.6%)	KX_FBK(23.9%)	ICL(21.4%)	UNICE(23.5%)
8	SEERLAB(22.0%)	Maui(23.9%)	DERIUNLP(20.1%)	SEERLAB(23.3%)
9	DFKI(21.7%)	DERIUNLP(23.6%)	DFKI(19.3%)	DFKI(22.2%)
10	Maui(19.3%)	UNPMC(22.6%)	BUAP(18.5%)	Maui(21.3%)
11	BUAP(18.5%)	SJTULTLAB(22.1%)	SJTULTLAB(17.9%)	DERIUNLP(20.3%)
12	JU_CSE(18.2%)	UNICE(21.8%)	JU_CSE(17.9%)	BUAP(19.7%)
13	Likey(18.2%)	DFKI(20.5%)	Maui(17.6%)	JU_CSE(18.6%)
14	SJTULTLAB(17.7%)	BUAP(20.2%)	UNPMC(17.6%)	UNPMC(17.8%)
15	UvT(15.8%)	UvT(20.2%)	UNICE(14.7%)	Likey(17.2%)
16	UNPMC(15.2%)	Likey(19.4%)	Likey(11.3%)	SJTULTLAB(16.7%)
17	UNIC(14.3%)	JU_CSE(17.3%)	POLYU(13.6%)	POLYU(14.3%)
18	POLYU(12.5%)	POLYU(15.8%)	UvT(10.3%)	UvT(12.6%)
19	UKP(4.4%)	UKP(5.0%)	UKP(5.4%)	UKP(6.8%)

Table 7: System ranking (and F-score) for each ACM classification: combined keywords

Rank	Group C	Group H	Group I	Group J
1	ICL(23.3%)	HUMB(25.0%)	HUMB(21.7%)	HUMB(24.7%)
2	KX_FBK(23.3%)	WINGNUS(23.5%)	KX_FBK(21.4%)	WINGNUS(24.4%)
3	HUMB(22.7%)	SEERLAB(23.2%)	SEERLAB(21.1%)	SZTERGAK(24.4%)
4	SZTERGAK(22.7%)	KP-Miner(22.4%)	WINGNUS(19.9%)	KX_FBK(24.4%)
5	DERIUNLP(21.5%)	SZTERGAK(21.8%)	KP-Miner(19.6%)	UNICE(23.8%)
6	KP-Miner(21.2%)	KX_FBK(21.2%)	SZTERGAK(19.6%)	ICL(23.5%)
7	WINGNUS(20.0%)	ICL(20.1%)	ICL(19.6%)	KP-Miner(22.6%)
8	SEERLAB(19.4%)	DERIUNLP(20.1%)	DFKI(18.5%)	SEERLAB(22.0%)
9	DFKI(19.4%)	DFKI(19.5%)	SJTULTLAB(17.6%)	DFKI(21.7%)
10	JU_CSE(17.0%)	SJTULTLAB(19.5%)	DERIUNLP(17.3%)	BUAP(19.6%)
11	Likey(16.4%)	UNICE(19.2%)	JU_CSE(16.7%)	DERIUNLP(19.0%)
12	SJTULTLAB(15.8%)	Maui(18.1%)	BUAP(16.4%)	Maui(17.8%)
13	BUAP(15.5%)	UNPMC(18.1%)	UNPMC(16.1%)	JU_CSE(17.9%)
14	Maui(15.2%)	Likey(16.9%)	Maui(14.9%)	Likey(17.5%)
15	UNICE(14.0%)	UvT(16.4%)	UNICE(14.0%)	UNPMC(16.6%)
16	UvT(14.0%)	POLYU(15.5%)	POLYU(11.9%)	SJTULTLAB(16.3%)
17	UNPMC(13.4%)	BUAP(14.9%)	Likey(10.4%)	POLYU(13.3%)
18	POLYU(12.5%)	JU_CSE(12.6%)	UvT(9.5%)	UvT(13.0%)
19	UKP(4.5%)	UKP(4.3%)	UKP(5.4%)	UKP(6.9%)

Table 8: System ranking (and F-score) for each ACM classification: reader-assigned keywords

Zhiyuan Liu and Peng Li and Yabin Zheng and Sun Maosong. Clustering to Find Exemplar Terms for Keyphrase Extraction. In *Proceedings of EMNLP*. 2009, pp. 257–266.

Yutaka Matsuo and Mitsuru Ishizuka. Keyword Extraction from a Single Document using Word Co-occurrence Statistical Information. *International Journal on Artificial Intelligence Tools*. 2004, 13(1), pp. 157–169.

Olena Medelyan and Ian H. Witten. Thesaurus based automatic keyphrase indexing. In *Proceedings of ACM/IEED-CS JCDL*. 2006, pp. 296–297.

Olena Medelyan. Human-competitive automatic topic indexing. PhD Thesis. University of Waikato. 2009.

Rada Mihalcea and Paul Tarau. TextRank: Bringing Order into Texts. In *Proceedings of EMNLP*. 2004, pp. 404–411.

Thuy Dung Nguyen and Min-Yen Kan. Key phrase Extraction in Scientific Publications. In *Proceedings of ICADL*. 2007, pp. 317–326.

Chau Q. Nguyen and Tuoi T. Phan. An ontology-based approach for key phrase extraction. In *Proceedings of the ACL-IJCNLP*. 2009, pp. 181–184.

Youngja Park and Roy J. Byrd and Branimir Boguraev. Automatic Glossary Extraction Beyond Terminology Identification. In *Proceedings of COLING*. 2004, pp. 48–55.

Peter Turney. Learning to Extract Keyphrases from Text. In *National Research Council, Institute for Information Technology, Technical Report ERB-1057*. 1999.

Peter Turney. Coherent keyphrase extraction via Web mining. In *Proceedings of IJCAI*. 2003, pp. 434–439.

Xiaojun Wan and Jianguo Xiao. CollabRank: towards a collaborative approach to single-document keyphrase extraction. In *Proceedings of COLING*. 2008, pp. 969–976.

Ian H. Witten and Gordon Paynter and Eibe Frank and Car Gutwin and Graig Nevill-Manning. KEA:Practical Automatic Key phrase Extraction. In *Proceedings of ACM conference on Digital libraries*. 1999, pp. 254–256.

Torsten Zesch and Iryna Gurevych. Approximate Matching for Evaluating Keyphrase Extraction. In *Proceedings of RANLP*. 2009.