Evaluating N-gram based Evaluation Metrics for Automatic Keyphrase Extraction

Su Nam Kim, Timothy Baldwin
CSSE
University of Melbourne
sunamkim@gmail.com, tb@ldwin.net

Min-Yen Kan
School of Computing
National University of Singapore
kanmy@comp.nus.edu.sg

Abstract

This paper describes a feasibility study of n-gram-based evaluation metrics for automatic keyphrase extraction. To account for near-misses currently ignored by standard evaluation metrics, we adapt various evaluation metrics developed for machine translation and summarization, and also the R-precision evaluation metric from keyphrase evaluation. In evaluation, the R-precision metric is found to achieve the highest correlation with human annotations. We also provide evidence that the degree of semantic similarity varies with the location of the partially-matching component words.

1 Introduction

Keyphrases are noun phrases (NPs) that are representative of the main content of documents. Since they represent the key topics in documents, extracting good keyphrases benefits various natural language processing (NLP) applications such as summarization, information retrieval (IR) and question-answering (QA). Keyphrases can also be used in text summarization as semantic metadata (Barzilay and Elhadad, 1997; Lawrie et al., 2001; D'Avanzo and Magnini, 2005). In search engines, keyphrases supplement full-text indexing and assist users in creating good queries.

In the past, a large body of work on keyphrases has been carried out as an extraction task, utilizing three types of cohesion: (1) document cohesion, i.e. cohesion between documents and keyphrases (Frank et al., 1999; Witten et al., 1999; Matsuo and Ishizuka, 2004; Medelyan and Witten, 2006; Nguyen and Kan, 2007; Wan and Xiao, 2008); (2) keyphrase cohesion, i.e. cohesion among keyphrases (Turney, 2003); and (3) term cohesion, i.e. cohesion among terms in a keyphrase (Park et al., 2004).

Despite recent successes in keyphrase extraction (Frank et al., 1999; Turney, 2003; Park et al., 2004; Medelyan and Witten, 2006; Nguyen and Kan, 2007), current work is hampered by the inflexibility of standard metrics in evaluating different approaches. As seen in other fields, e.g. machine translation (MT) and multi-document summarization, the advent of standardized automatic evaluation metrics, combined with standardized datasets, has enabled easy comparison of systems and catalyzed the respective research areas. Traditionally, the evaluation of automatic keyphrase extraction has relied on the number of exact matches in author-assigned keyphrases and reader-assigned keyphrases. The main problem with this approach is that even small variants in the keyphrases are not given any credit. For example, given the gold-standard keyphrase *effective grid computing algorithm*, *grid computing algorithm* is a plausible keyphrase candidate and should be scored appropriately, rather than being naively evaluated as wrong. Additionally, author-assigned keyphrases and even reader-assigned keyphrases often have their own problems in this type of evaluation (Medelyan and Witten, 2006). For example, some keyphrases are often partly or wholly subsumed by other candidates or may not even occur in the document. Therefore, counting the exactly-matching candidates has been shown to be suboptimal (Jarmasz

Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), pages 572–580,
Beijing, August 2010

and Barriere, 2004).

Our goal in this paper is to evaluate the reliability of automatic evaluation metrics that better account for near-misses. Prior research based on semantic similarity (Jarmasz and Barriere, 2004; Mihalcea and Tarau, 2004; Medelyan and Witten, 2006) has taken the approach of using external resources such as large corpora, Wikipedia or manually-curated index words. While we acknowledge that these methods can help address the near-miss problem, they are impractical due to the effort required to compile the requisite resources for each individual evaluation exercise, and furthermore, the resources tend to be domain-specific. In order to design a cheap, practical and stable keyphrase evaluation metric, our aim is to properly account for these near-misses without reliance on costly external resources.

According to our analysis, the degree of semantic similarity of keyphrase candidates varies relative to the location of overlap. For example, the candidate *grid computing algorithm* has higher semantic similarity than *computing algorithm* with the gold-standard keyphrase *effective grid computing algorithm*. Also, *computing algorithm* is closer than *effective grid* to the same gold-standard keyphrase. From these observations, we infer that n-gram-based evaluation metrics can be applied to evaluating keyphrase extraction, but also that candidates with the same relative n-gram overlap are not necessarily equally good.

Our primary goal is to test the utility of n-gram based evaluation metrics to the task of keyphrase extraction evaluation. We test the following evaluation metrics: (1) evaluation metrics from MT and multi-document summarization (BLEU, NIST, METEOR and ROUGE); and (2) R-precision (Zesch and Gurevych, 2009), an n-gram-based evaluation metric developed specifically for keyphrase extraction evaluation which has yet to be evaluated against humans at the extraction task. Secondarily, we attempt to shed light on the bigger question of whether it is feasible to expect that n-gram-based metrics without access to external resources should be able to capture subtle semantic differences in keyphrase candidates. To this end, we experimentally verify the impact of lexical overlap of different types on keyphrase similarity, and use this as the basis for proposing a variant of R-precision.

In the next section, we present a brief primer on keyphrases. We then describe the MT and summarization evaluation metrics trialled in this research, along with R-precision, modified R-precision and a semantic similarity-based evaluation metric for keyphrase evaluation (Section 3). In Section 4, we discuss our gold-standard and candidate extraction method. We compare the evaluation metrics with human assigned scores for suitability in Section 5, before concluding the paper.

2 A Primer on Keyphrases

Keyphrases can be either simplex words (e.g. *query*, *discovery*, or *context-awareness*)[1] or larger N-bars/noun phrases (e.g. *intrusion detection*, *mobile ad-hoc network*, or *quality of service*). The majority of keyphrases are 1–4 words long (Paukkeri et al., 2008).

Keyphrases are normally composed of nouns and adjectives, but may occasionally contain adverbs (e.g. *dynamically allocated task*, or *partially observable Markov decision process*) or other parts of speech. They may also contain hyphens (e.g. *sensor-grouping* or *multi-agent system*) and apostrophes for possessives (e.g. *Bayes' theorem* or *agent's goal*).

Keyphrases can optionally incorporate PPs (e.g. *service quality* vs. *quality of service*). A variety of prepositions can be used (e.g. *incentive for cooperation, inequality in welfare, agent security via approximate policy*), although the genetive *of* is the most common.

Keyphrases can also be coordinated, either as simple nouns at the top level (e.g. *performance and scalability* or *group and partition*) or within more complex NPs or between N-bars (e.g. *history of past encounter and transitivity* or *task and resource allocation in agent system*).

When candidate phrases get too long, abbreviations also help to form valid keyphrases (e.g. *computer support collaborative work* vs. *CSCW*, or *partially observable Markov decision process* vs. *POMDP*).

[1] All examples in this section are taken from the data set outlined in Section 4.

3 Evaluation Metrics

There have been various evaluation metrics developed and validated for reliability in fields such as MT and summarization (Callison-Burch et al., 2009). While n-gram-based metrics don't capture systematic alternations in keyphrases, they do support partial match between keyphrase candidates and the reference keyphrases.

In this section, we first introduce a range of popular n-gram-based evaluation metrics from the MT and automatic summarization literature, which we naively apply to the task of keyphrase evaluation. We then present R-precision, an n-gram-based evaluation metric developed specifically for keyphrase evaluation, and propose a modified version of R-precision which weights n-grams according to their relative position in the keyphrase. Finally, we present a semantic similarity method.

3.1 Machine Translation and Summarization Evaluation Metrics

In this research, we experiment with four popular n-gram-based metrics from the MT and automatic summarization fields — BLEU, METEOR, NIST and ROUGE. The basic task performed by the respective evaluation metrics is empirical determination of *how good an approximation is string*$_1$ *of string*$_2$*?*, which is not far removed from the requirements of keyphrase evaluation. We briefly outline each of the methods below.

One subtle property of keyphrase evaluation is that there is no a priori preference for shorter keyphrases over longer keyphrases, unlike MT where shorter strings tend to be preferred. Hence, we use the longer NP as reference and the shorter NP as a translation, to avoid the length penalty in most MT metrics.[2]

BLEU (Papineni et al., 2002) is an evaluation metric for measuring the relative similarity between a candidate translation and a set of reference translations, based on n-gram composition. It calculates the number of overlapping n-grams between the candidate translation and the set of reference translations. In order to avoid having very short translations receive artificially high scores, BLEU adds a brevity penalty to the scoring equation.

METEOR (Agarwal and Lavie, 2008) is similar to BLEU, in that it measures string-level similarity between the reference and candidate translations. The difference is that it allows for more match flexibility, including stem variation and WordNet synonymy. The basic metric is based on the number of mapped unigrams found between the two strings, the total number of unigrams in the translation, and the total number of unigrams in the reference.

NIST (Martin and Przybocki, 1999) is once again similar to BLEU, but integrates a proportional difference in the co-occurrences for all n-grams while weighting more heavily n-grams that occur less frequently, according to their information value.

ROUGE (Lin and Hovy, 2003) — and its variants including ROUGE-N and ROUGE-L — is similarly based on n-gram overlap between the candidate and reference summaries. For example, ROUGE-N is based on co-occurrence statistics, using higher-order n-grams ($n > 1$) to estimate the fluency of summaries. ROUGE-L uses longest common subsequence (LCS)-based statistics, based on the assumption that the longer the substring overlap between the two strings, the greater the similar Saggion et al. (2002). ROUGE-W is a weighted LCS-based statistic that prioritizes consecutive LCSes. In this research, we experiment exclusively with the basic ROUGE metric, and unigrams (i.e. ROUGE-1).

3.2 R-precision

In order to analyze near-misses in keyphrase extraction evaluation, Zesch and Gurevych (2009) proposed R-precision, an n-gram-based evaluation metric for keyphrase evaluation.[3] R-precision contrasts with the majority of previous work on keyphrase extraction evaluation, which has used semantic similarity based on external resources

[2] While we don't present the numbers in this paper, the results were lower for the MT evaluation metrics without this reordering of the reference and candidate keyphrases.

[3] Zesch and Gurevych's R-precision has nothing to do with the information retrieval evaluation metric of the same name, where P@N is calculated for N equal to the number of relevant documents.

(Jarmasz and Barriere, 2004; Mihalcea and Tarau, 2004; Medelyan and Witten, 2006). As our interest is in fully automated evaluation metrics which don't require external resources and are domain independent (for maximal reproducibility of results), we experiment only with R-precision in this paper.

R-precision is based on the number of overlapping words between a keyphrase and a candidate, as well as the length of each. The metric differentiates three types of near-misses: *INCLUDE*, *PARTOF* and *MORPH*. The first two types are based on an n-gram approach, while the third relies on lexical variation. As we use stemming, in line with the majority of previous work on keyphrase extraction evaluation, we focus exclusively on the first two cases, namely *INCLUDE*, and *PARTOF*. The final score returned by R-precision is:

$$\frac{\text{number of overlapping word(s)}}{\text{length of keyphrase/candidate}}$$

where the denominator is the longer of the keyphrase and candidate.

Zesch and Gurevych (2009) evaluated R-precision over three corpora (Inspec, DUC and SP) based on 566 non-exact matching candidates. In order to evaluate the human agreement, they hired 4 human annotators to rate the near-miss candidates, and reported agreements of 80% and 44% for the INCLUDE and PARTOF types, respectively. They did not, however, perform holistic evaluation with human scores to verify its reliability in full system evaluation. This is one of our contributions in this paper.

3.3 Modified R-precision

In this section, we describe a modification to R-precision which assigns different weights for component words based on their position in the keyphrase (unlike R-precision which assigns the same score for each matching component word). The head noun generally encodes the core semantics of the keyphrase, and as a very rough heuristic, the further a word is from the head noun, the less semantic import on the keyphrase it has. As such, modified R-precision assigns a score to each component word relative to its position as $CW = \frac{1}{N-i+1}$ where N is the number of component words in the keyphrase and i is the position of the component word in the keyphrase (1 = leftmost word).

For example, *AB* and *BC* from *ABC* would be scored as $\frac{\frac{1}{3}+\frac{1}{2}}{\frac{1}{3}+\frac{1}{2}+\frac{1}{1}} = \frac{5}{11}$ and $\frac{\frac{1}{2}+\frac{1}{1}}{\frac{1}{3}+\frac{1}{2}+\frac{1}{1}} = \frac{9}{11}$, respectively. Thus, with the keyphrase *effective grid computing algorithm* and candidates *effective grid*, *grid computing* and *computing algorithm*, modified R-precision assigns different scores for each candidate (*computing algorithm* $>$ *grid computing* $>$ *effective grid*). In contrast, the original R-precision assigns the same score to all candidates.

3.4 Semantic Similarity

In Jarmasz and Barriere (2004) and Mihalcea and Tarau (2004), the authors used a large data set to compute the semantic similarity of two NPs to assign partial credits for semantically similar candidate keyphrases. To simulate these methods, we adopted the distributional semantic similarity using web documents. That is, we computed the similarity between a keyphrase and its substring by cosine measure over collected the snippets from Yahoo! BOSS.[4] We use the computed similarity as our score for near-misses.

4 Data

4.1 Data Collection

We constructed a keyphrase extraction dataset using papers across 4 different categories[5] of the ACM Digital Library.[6] In addition to author-assigned keyphrases provided as part of the ACM Digital Library, we generated reader-assigned keyphrases by assigning 250 students 5 papers each, a list of candidate keyphrases (see below for details), and standardized instructions on how to assign keyphrases. It took them an average of 15 minutes to annotate each paper. This is the same

[4] http://developer.yahoo.com/search/boss/

[5] C2.4 (Distributed Systems), H3.3 (Information Search and Retrieval), I2.11 (Distributed Artificial Intelligence – Multiagent Systems) and J4 (Social and Behavioral Sciences – Economics).

[6] http://portal.acm.org/dl.cfm

	Author	Reader	Total
Total	1298/1305	3110/3221	3816/3962
NPs	937	2537	3027
Average	3.85/4.01	12.44/12.88	15.26/15.85
Found	769	2509	2864

Table 1: Details of the keyphrase dataset

(**Rule1**) NBAR = `(NN*|JJ*)`*`(NN*)`
e.g. *complexity, effective algorithm,*
distributed web-service discovery architecture
(**Rule2**) NBAR `IN` NBAR
e.g. *quality of service, sensitivity of VOIP traffic,*
simplified instantiation of zebroid

Table 2: Regular expressions for candidate selection

document collection and set of keyphrase annotations as was used in the SemEval 2010 keyphrase extraction task (Kim et al., 2010).

Table 1 shows the details of the final dataset. The numbers after the slashes indicate the number of keyphrases after including alternate keyphrases based on *of*-PPs. Despite the reliability of author-assigned keyphrases discussed in Medelyan and Witten (2006), many author-assigned keyphrases and some reader-assigned keyphrases are not found verbatim in the source documents because: (1) many of them are substrings of the candidates or vice versa (about 75% of the total keyphrases are found in the documents); and (2) our candidate selection method does not extract keyphrases in forms such as coordinated NPs or adverbial phrases.

4.2 Candidate Selection

During preprocessing, we first converted the PDF versions of the papers into text using `pdftotext`. We then lemmatized and POS tagged all words using `morpha` and the `Lingua` POS tagger. Next, we applied the regular expressions in Table 2 to extract candidates, based on Nguyen and Kan (2007). Finally, we selected candidates in terms of their frequency: simplex words with frequency ≥ 2 and NPs with frequency ≥ 1. We observed that for reader-assigned keyphrases, NPs were often selected regardless of their frequency in the source document. In addition, we allowed variation in the possessive form, noun number and abbreviations.

Rule1 detects simplex nouns or N-bars as candidates. *Rule2* extracts N-bars with post-modifying PPs. In Nguyen and Kan (2007), *Rule2* was not used to additionally extract N-bars inside modifying PPs. For example, our rules extract not only *performance of grid computing* as a candidate, but also *grid computing*. However, we did not extend the candidate selection rules to cover NPs including adverbs (e.g. *partially-observable Markov decision process*) or conjunctions (e.g. *behavioral evolution and extrapolation*), as they are rare.

4.3 Human Assigned Score

We hired four graduate students working in NLP to assign human scores to substrings in the gold-standard data. The scores are between 0 and 4 (0 means no semantic overlap between a NP and its substring, while 4 means semantically indistinguishable).

We broke down the candidate–keyphrases pairs into subtypes, based on where the overlap occurs relative to the keyphrase (e.g. *ABCD*): (1) *Head*: the candidate contains the head noun of the keyphrase (e.g. *CD*); (2) *First*: the candidate contains the first word of the keyphrase (e.g. *AB*); and (3) *Middle*: the candidate overlaps with the keyphrase, but contains neither its first word nor its head word (e.g. *BC*). The average human scores are 1.94 and 2.11 for *First* and *Head*, respectively, when the candidate is shorter, while they are 2.00, 1.89 and 2.15 for *First*, *Middle*, and *Head*, respectively when the candidate is longer. Note that we did not have *Middle* instances with candidates as the shorter string. The scores are slightly higher for the keyphrases as substrings than for the candidates as substrings.

5 Correlation

To check the feasibility of metrics for keyphrase evaluation, we checked the Spearman rank correlation between the machine-generated score and the human-assigned score for each keyphrase–candidate pairing.

As the percentage of annotators who agree on the exact score is low (i.e. 2 subjects agree ex-

		Human	R-precision		BLEU	METEOR	NIST	ROUGE	Semantic Similarity
			Orig	Mod					
Average	All	.4506	.4763	.2840	.3250	.3246	.3366	.3246	.2116
	$L \leq 4$	.4510	.5264	.2806	.3242	.3238	.3369	.3240	.2050
	$L \leq 3$	.4551	.4834	.2893	.3439	.3437	.3584	.3437	.1980
Majority	All	.4603	.4763	.3438	.3407	.3403	.3514	.3404	.2224
	$L \leq 4$	.4604	.5264	.3434	.3423	.3421	.3547	.3422	.2168
	$L \leq 3$	.4638	.4838	.3547	.3679	.3675	.3820	.3676	.2123

Table 3: Rank correlation between humans and the different evaluation metrics, based on the human average (top half) and majority (bottom half)

		Human	R-precision		BLEU	METEOR	NIST	ROUGE
			Orig	Mod				
LOCATION	First	**.5508**	**.5032**	**.5033**	.3844	.3844	.4057	.3844
	Middle	**.5329**	**.5741**	**.5988**	**.4669**	**.4669**	.4055	**.4669**
	Head	**.3783**	**.4838**	**.4838**	.3865	.3860	.3780	.3864
COMPLEXITY	Simple	.4452	**.4715**	.2790	.3653	.3445	.3527	.3445
	PP	**.4771**	**.4814**	.1484	.3367	.3122	.3443	.3123
	CC	.3645	.3810	.3140	.3748	.3446	.3384	.3748
POS	AdjN	**.4616**	**.4844**	.3507	.3147	.3132	.3115	.3133
	NN	.4467	**.4586**	.2581	.3321	.3321	.3488	.3322

Table 4: Rank correlation between human average judgments and n-gram-based metrics

actly on 55%-70% of instances, 3 subjects agree exactly on 25%-35% of instances), we require a method for combining the annotations. We experiment with two combination methods: majority and average. The majority is simply the label with the majority of annotations associated with it; in the case of a tie, we break the tie by selecting that annotation which is closest to the median. The average is simply the average score across all annotators.

5.1 Overall Correlation with Human Scores

Table 3 presents the correlations between the human scores (acting as an upper bound for the task), as well as those between human scores with machine-generated scores. We first present the overall results, then results over the subset of keyphrases of length 4 words or less, and also 3 words or less. We present the results for the annotator average and majority in top and bottom half, respectively, of the table.

To compute the correlation between the human annotators, we used leave-one-out cross-validation, holding out one annotator, and comparing them to the combination of the remaining annotators (using either the majority or average method to combine the remaining annotations). This was repeated across all annotators, and the Spearman's ρ was averaged across the annotators.

Overall, we found that R-precision achieved the highest correlation with humans, above the inter-annotator correlation in all instances. That is, based on the evaluation methodology employed, it is performing slightly above the average level of a single annotator. The relatively low inter-annotator correlation is, no doubt, due to the difficulty of the task, as all of our near-misses have 2 or more terms, and the annotators have to make very fine-grained, and ultimately subjective, decisions about the true quality of the candidate.

Comparing the n-gram-based methods with the semantic similarity-based method, the n-gram-based metrics achieved higher correlations across the board, with BLEU, METEOR, NIST and ROUGE all performing remarkably consistently, but well

		Human	R-precision		BLEU	METEOR	NIST	ROUGE
			Orig	Mod				
LOCATION	First	**.5642**	**.5162**	**.5163**	.4032	.4032	.4297	.4032
	Middle	**.5510**	**.4991**	**.5320**	.4175	.4175	.3653	.4175
	Head	.4147	**.5073**	**.5074**	.4156	.4153	.4042	.4156
COMPLEXITY	Simple	.4580	**.4869**	.3394	.3653	.3651	.3715	.3651
	PP	**.4715**	**.5068**	.3724	.3367	.3367	.3652	.3367
	CC	**.5777**	**.5513**	.3841	**.5745**	**.5571**	**.5600**	**.5745**
POS	AdjN	.4501	**.4861**	.3968	.3266	.3251	.3246	.3252
	NN	**.4631**	**.4733**	.3244	.3499	.3499	.3648	.3500

Table 5: Rank correlation between human majority and n-gram-based metrics

below the level of R-precision. Due to the markedly lower performance of the semantic similarity-based method, we do not consider it for the remainder of our experiments. A general finding was that as the length of the keyphrase (L) got longer, the correlation tended to be higher across all n-gram-based metrics.

One disappointment at this stage is that the results for modified R-precision are well below those of the original, especially over the average of the human annotators.

5.2 Correlation with Different NP Subtypes

To get a clearer sense of how the different evaluation metrics are performing, we broke down the keyphrases according to three syntactic subclassifications: (1) the location of overlap (see Section 4.3); (2) the complexity of the NP (does the keyphrase contain a preposition [PP], a conjunction [CC] or neither a preposition nor a conjunction [Simple]?); and (3) the word class sequence of the keyphrase (is the keyphrase an NN [NN] or an AdjN sequence [AdjN]?). We present the results in Tables 4 and Table 4 for the human average and majority, respectively, presenting results in **boldface** when the correlation for a given method is higher than for that same method in our holistic evaluation in Table 3 (i.e. .4506 and .4603, for the average and majority human scores, respectively).

All methods, including inter-annotator correlation, improve in raw numbers over the subsets of the data based on overlap location, indicating that the data was partitioned into more internally-consistent subsets. Encouragingly, modified R-precision equalled or bettered the performance of the original R-precision over each subset of the data based on overlap location. Where modified R-precision appears to fall down most noticeably is over keyphrases including prepositions, as our assumption about the semantic import based on linear ordering clearly breaks down in the face of post-modifying PPs. It is also telling that it does worse over noun–noun sequences than adjective–noun sequences. In being agnostic to the effects of syntax, the original R-precision appears to benefit overall. Another interesting effect is that the performance of BLEU, METEOR and ROUGE is notably better over candidates which match with non-initial and non-final words in the keyphrase.

We conclude from this analysis that keyphrase scoring should be sensitive to overlap location. Furthermore, our study also shows that n-gram-based MT and summarization metrics are surprisingly adept at capturing partial matches in keyphrases, despite them being much shorter than the strings they are standardly applied to. More compellingly, we found that R-precision is the best overall performer, and that it matches the performance of our human annotators across the board. This is the first research to establish this fact. Our findings for modified R-precision were more sobering, but its location sensitivity was shown to improve over R-precision for instances of overlap in the middle or with the head of the keyphrase.

6 Conclusion

In this work, we have shown that preexisting n-gram-based evaluation metrics from MT, summarization and keyphrase extraction evaluation are able to handle the effects of near-misses, and that R-precision performs at or above the average level of a human annotator. We have also shown that a semantic similarity-based method which uses web data to model distributional similarity performed below the level of all of the n-gram-based methods, despite them requiring no external resources (web or otherwise). We proposed a modification to R-precision based on the location of match, but found that while it could achieve better performance over certain classes of keyphrases, its net effect was to drag the performance of R-precision down. Other methods were found to be remarkably consistent across different subtypes of keyphrase.

Acknowledgements

Many thanks to the anonymous reviewers for their insightful comments. We wish to acknowledge the generous funding from National Research Foundation grant R 252-000-279-325 in supporting Min-Yen Kan's work.

References

Abhaya Agrwal and Alon Lavie. METEOR, M-BLEU and M-TER: Evaluation Metrics for High-Correlation with Human Rankings of Machine Translation Output. In *Proceedings of ACL Workshop on Statistical Machine Translation*. 2008.

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.

Chris Callison-Burch, Philipp Koehn, Christof Monz and Josh Schroeder. Proceedings of 4th Workshop on Statistical Machine Translation. 2009.

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

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

Su Nam Kim, Olena Medelyan, Min-Yen Kan and Timothy Baldwin. SemEval-2010 Task 5: Automatic Keyphrase Extraction from Scientific Articles. In *Proceedings of SemEval-2: Evaluation Exercises on Semantic Evaluation*. to appear.

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

Chin-Yew Lin and Edward H. Hovy. Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics. In *In Proceedings of HLT-NAACL*. 2003.

Alvin Martin and Mark Przybocki. The 1999 NIST Speaker Recognition Evaluation, Using Summed Two-Channel Telephone Data for Speaker Detection and Speaker Tracking. In *Proceedings of EuroSpeech*. 1999.

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

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

Guido Minnen, John Carroll and Darren Pearce. Applied morphological processing of English. *NLE*. 2001, 7(3), pp. 207–223.

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

Sebastian Padó, Michel Galley, Dan Jurafsky and Christopher D. Manning. Textual Entailment Features for Machine Translation Evaluation. In *Proceedings of ACL Workshop on Statistical Machine Translation*. 2009, pp. 37–41.

Kishore Papineni, Salim Roukos, Todd Ward and Wei-Jing Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of ACL*. 2001, pp. 311–318.

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

Mari-Sanna Paukkeri, Ilari T. Nieminen, Matti Polla and Timo Honkela. A Language-Independent Approach to Keyphrase Extraction and Evaluation. In *Proceedings of COLING*. 2008, pp. 83–86.

Horacio Saggion, Dragomir Radev, Simon Teufel, Wai Lam and Stephanie Strassel. Meta-evaluation of Summaries in a Cross-lingual Environment using Content-based Metrics. In *Proceedings of COLING*. 2002, pp. 1–7.

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 Witten, Gordon Paynter, Eibe Frank, Car Gutwin and Craig 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. *International Conference on Recent Advances in Natural Language Processing*. 2009.