4 Evaluation

4.1 Experimental Setup

TextRank and SingleRank setup Following Mihalcea and Tarau (2004) and Wan and Xiao (2008), we set the co-occurrence window size for TextRank and SingleRank to 2 and 10, respectively, as these parameter values have yielded the best results for their evaluation datasets.

ExpandRank setup Following Wan and Xiao (2008), we find the 5 nearest neighbors for each document from the remaining documents in the same corpus. The other parameters are set in the same way as in SingleRank.

KeyCluster setup As argued by Liu et al. (2009b), Wikipedia-based relatedness is computationally expensive to compute. As a result, we follow them by computing the *co-occurrence-based* relatedness instead, using a window of size 10. Then, we cluster the candidate words using spectral clustering, and use the frequent word list that they generously provided us to post-process the resulting keyphrases by filtering out those that are frequent unigrams.

4.2 Results and Discussion

In an attempt to gain a better insight into the five unsupervised systems, we report their performance in terms of precision-recall curves for each of the four datasets (see Figure 1). This contrasts with essentially all previous work, where the performance of a keyphrase extraction system is reported in terms of an F-score obtained via a particular parameter setting on a particular dataset. We generate the curves for each system as follows. For Tf-Idf, SingleRank, and ExpandRank, we vary the number of keyphrases, N, predicted by each system. For TextRank, instead of varying the number of predicted keyphrases, we vary T, the percentage of top-scored vertices (i.e., unigrams) that are selected as keywords at the end of the ranking step. The reason is that TextRank only imposes a ranking on the unigrams but not on the keyphrases generated from the high-ranked unigrams. For KeyCluster, we vary the number of clusters produced by spectral clustering rather than the number of predicted keyphrases, again because KeyCluster does not impose a ranking on the resulting keyphrases. In addition, to give an estimate of how each system performs in terms of F-score, we also plot curves corresponding to different F-scores in these graphs.

Tf-Idf Consistent with our intuition, the precision of Tf-Idf drops as recall increases. Although it is the simplest of the five approaches, Tf-Idf is the best performing system on all but the *Inspec* dataset, where TextRank and KeyCluster beat Tf-Idf on just a few cases. It clearly outperforms all other systems for NUS and ICSI.

TextRank The TextRank curves show a different progression than Tf-Idf: precision does not drop as much when recall increases. For instance, in case of DUC and ICSI, precision is not sensitive to changes in recall. Perhaps somewhat surprisingly, its precision increases with recall for *Inspec*, allowing it to even reach a point (towards the end of the curve) where it beats Tf-Idf. While additional experiments are needed to determine the reason for this somewhat counter-intuitive result, we speculate that this may be attributed to the fact that the TextRank curves are generated by progressively increasing T (i.e., the percentage of top-ranked vertices/unigrams that are used to generate keyphrases) rather than the number of predicted keyphrases, as mentioned before. Increasing T does not necessarily imply an increase in the number of predicted keyphrases, however. To see the reason, consider an example in which we want TextRank to extract the keyphrase "advanced machine learning" for a given document. Assume that TextRank ranks the unigrams "advanced", "learning", and "machine" first, second, and third, respectively in its ranking step. When $T = \frac{2}{n}$, where n denotes the total number of candidate unigrams, only the two highest-ranked unigrams (i.e., "advanced" and "learning") can be used to form keyphrases. This implies that "advanced" and "learning" will each be predicted as a keyphrase, but "advanced machine learning" will not. However, when $T = \frac{3}{n}$, all three unigrams can be used to form a keyphrase, and since TextRank collapses unigrams adjacent to each other in the text to form a keyphrase, it will correctly predict "advanced machine learning" as a keyphrase. Note that as we increase T from $\frac{2}{n}$ to $\frac{3}{n}$, recall increases, and so does precision, since