jor directions in this research area, including Tf-Idf and four recently proposed systems, namely, TextRank (Mihalcea and Tarau, 2004), SingleRank (Wan and Xiao, 2008), ExpandRank (Wan and Xiao, 2008), and a clustering-based approach (Liu et al., 2009b). Since none of these systems (except TextRank) are publicly available, we reimplement all of them and make them freely available for research purposes.[1] To our knowledge, this is the *first* attempt to compare the performance of state-of-the-art unsupervised keyphrase extraction systems on multiple datasets.

## 2 Corpora

Our four evaluation corpora belong to different domains with varying document properties. Table 1 provides an overview of each corpus.

The **DUC-2001** dataset (Over, 2001), which is a collection of 308 news articles, is annotated by Wan and Xiao (2008). We report results on all 308 articles in our evaluation.

The **Inspec** dataset is a collection of 2,000 abstracts from journal papers including the paper title. Each document has two sets of keyphrases assigned by the indexers: the *controlled keyphrases*, which are keyphrases that appear in the *Inspec* thesaurus; and the *uncontrolled keyphrases*, which do not necessarily appear in the thesaurus. This is a relatively popular dataset for automatic keyphrase extraction, as it was first used by Hulth (2003) and later by Mihalcea and Tarau (2004) and Liu et al. (2009b). In our evaluation, we use the set of 500 abstracts designated by these previous approaches as the test set and its set of uncontrolled keyphrases. Note that the average document length for this dataset is the smallest among all our datasets.

The **NUS Keyphrase Corpus** (Nguyen and Kan, 2007) includes 211 scientific conference papers with lengths between 4 to 12 pages. Each paper has one or more sets of keyphrases assigned by its authors and other annotators. We use all the 211 papers in our evaluation. Since the number of annotators can be different for different documents and the annotators are not specified along with the annotations, we decide to take the union of all the gold standard keyphrases from all the sets to construct one single set of annotation for each paper. As Table 1 shows, each NUS paper, both in terms of the average number of tokens (8291) and candidate phrases (2027) per paper, is more than five times larger than any document from any other corpus. Hence, the number of candidate keyphrases that can be extracted is potentially large, making this corpus the most challenging of the four.

Finally, the **ICSI meeting corpus** (Janin et al., 2003), which is annotated by Liu et al. (2009a), includes 161 meeting transcriptions. Following Liu et al., we remove topic segments marked as 'chitchat' and 'digit' from the dataset and use all the remaining segments for evaluation. Each transcript contains three sets of keyphrases produced by the same three human annotators. Since it is possible to associate each set of keyphrases with its annotator, we evaluate each system on this dataset three times, once for each annotator, and report the average score. Unlike the other three datasets, the gold standard keys for the ICSI corpus are mostly unigrams.

## 3 Unsupervised Keyphrase Extractors

A generic unsupervised keyphrase extraction system typically operates in three steps (Section 3.1), which will help understand the unsupervised systems explained in Section 3.2.

### 3.1 Generic Keyphrase Extractor

**Step 1: Candidate lexical unit selection** The first step is to filter out unnecessary word tokens from the input document and generate a list of potential keywords using heuristics. Commonly used heuristics include (1) using a stop word list to remove non-keywords (e.g., Liu et al. (2009b)) and (2) allowing words with certain part-of-speech tags (e.g., nouns, adjectives, verbs) to be considered candidate keywords (Mihalcea and Tarau (2004), Liu et al. (2009a), Wan and Xiao (2008)). In all of our experiments, we follow Wan and Xiao (2008) and select as candidates words with the following Penn Treebank tags: NN, NNS, NNP, NNPS, and JJ, which are obtained using the Stanford POS tagger (Toutanova and Manning, 2000).

[1] See `http://www.hlt.utdallas.edu/~saidul/code.html` for details.