Topic Indexing with Wikipedia

Olena Medelyan, Ian H. Witten and David Milne

Computer Science Department
University of Waikato
New Zealand
{olena, ihw, dnk2}@cs.waikato.ac.nz

Abstract

Wikipedia article names can be utilized as a controlled vocabulary for identifying the main topics in a document. Wikipedia's 2M articles cover the terminology of nearly any document collection, which permits controlled indexing in the absence of manually created vocabularies. We combine state-of-the-art strategies for automatic controlled indexing with Wikipedia's unique property—a richly hyperlinked encyclopedia. We evaluate the scheme by comparing automatically assigned topics with those chosen manually by human indexers. Analysis of indexing consistency shows that our algorithm outperforms some human subjects.

1. Introduction

The main topics of a document often indicate whether or not it is worth reading. In libraries of yore, professional human indexers were employed to manually categorize documents, and the result was offered to users along with other metadata. However, the explosion of information has made it infeasible to sustain such a labor-intensive process.

Automated indexing has been investigated from various angles. *Keyphrase extraction* weights word n-grams or syntactic phrases that appear in a document according to their statistical properties. The resulting index terms are restricted to phrases that occur in the document, and are prone to error because semantic relations are ignored. *Term assignment* uses text classification to create a model for every topic against which new documents are compared; but this needs a huge volume of training data. The inaccuracy of keyphrase extraction and the impracticality of term assignment have stimulated a new method, *keyphrase indexing*, which maps document phrases into related terms of a controlled vocabulary that do not necessarily appear verbatim, and weights terms based on certain features. Problems of ambiguity and the need for a manually created vocabulary restrict the technique to narrow domains.

The online encyclopedia Wikipedia is tantamount to a huge controlled vocabulary whose structure and features resemble those of thesauri, which are commonly used as indexing vocabularies—as illustrated by Figure 1 (Milne *et al.* 2006). Wikipedia articles (and redirects) correspond to terms. Its extensive coverage makes Wikipedia applicable to nearly any domain. However, its vast size creates new challenges when mapping documents to Wikipedia articles.

This paper shows how Wikipedia can be utilized effectively for topical indexing. The scheme is evaluated on a set of 20 computer science articles, indexed by 15 teams of computer science students working independently, two per team. The automatic approach outperforms some student teams, and needs only a very small training set.

2. Related work

One of the largest controlled vocabularies used for indexing is the Medical Subject Heading (MeSH) thesaurus. It contains 25,000 concepts and has been applied to both term assignment and keyphrase indexing, individually and in combination. Markó *et al.* (2004) decompose document phrases into morphemes with a manually created dictionary and associate them with MeSH terms assigned to the documents. After training on 35,000 abstracts they assign MeSH terms to unseen documents with precision and recall of around 30% for the top 10 terms. However, only concepts that appear in the training data can be assigned to new documents, and the training corpus must be large.

Aronson *et al.* (2000) decompose candidate phrases into letter trigrams and use vector similarity to map them to concepts in the UMLS thesaurus. The UMLS structure allows these concepts to be converted to MeSH terms. The candidates are augmented by additional MeSH terms from the 100 closest documents in the manually indexed PubMed collection, and the terms are heuristically weighted. An experiment with 500 full text articles achieved 60% recall and 31% precision for the top 25 terms (Gay *et al.*, 2005). However, the process seems to involve the entire PubMed corpus, millions of manually indexed documents.

The key challenge is overcoming terminological differences between source documents and vocabulary terms. Wikipedia, with over 2M synonyms for terms ("redirects"), extensively addresses spelling variations, grammatical variants and synonymy. The 4.7M anchor links offer additional clues to how human contributors refer to articles.

A second issue is the need for large amounts of training data in both the systems mentioned above. In contrast, Medelyan and Witten (2008) achieve good results with fewer than 100 training documents by learning typical properties of manually assigned terms in general, instead of associations between particular index terms and docu-

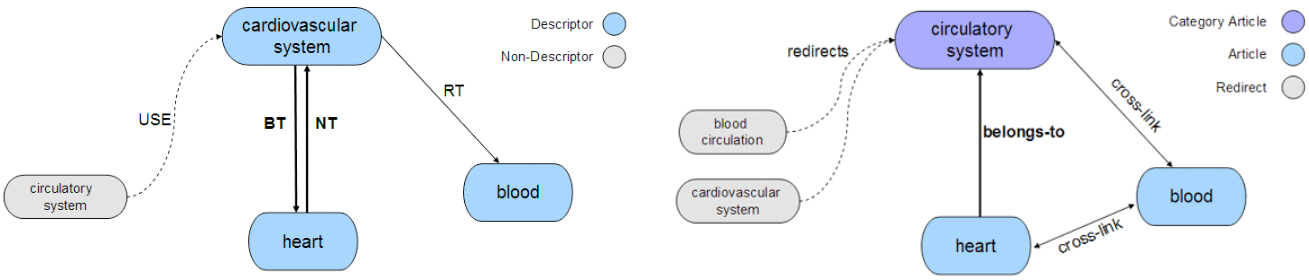

Figure 1. Excerpts from manually created Agrovoc thesaurus and thesaurus automatically built from Wikipedia

ment phrases. To ensure semantic conflation they use synonymy links encoded in a manual thesaurus. Each candidate phrase is characterized by several features (see Section 3.4 below). A Naïve Bayes scheme is used to learn a model which is applied to unseen documents. Performance improves if "degree of correctness" data is available from multiple indexers: use the number of indexers who choose a term as a keyphrase instead of whether or not one indexer has been chosen it. The method yields 32% consistency with professional indexers, compared with a figure of 39% for human indexers. It is domain-independent but requires a manually created controlled vocabulary.

In this paper we distil a controlled vocabulary automatically from Wikipedia. Wikipedia has been used for similar tasks before. Gabrilovich and Markovich (2007) improve text classification by adding information from it to the bag-of-words document model. They build a vector space model of all Wikipedia articles, and, before classifying a new document, site it in the space and add the most similar articles' titles as new features. However, documents are classified into only a few hundred categories, whereas our goal is treat every Wikipedia article as a potential topic.

Mihalcea and Csomai (2007) describe the new problem of "wikification," which has similarities to topic indexing. A document is "wikified" by linking it to Wikipedia articles in a way that emulates how a human contributor might add a new article to Wikipedia. For each n-gram that appears in Wikipedia they pre-compute the probability of it being a link—they call this its "keyphraseness." Then all phrases in a new document whose keyphraseness exceeds a certain threshold are chosen as keyphrases.

Their usage of the term "keyphrase" diverges from the conventional meaning. Keyphrases are words or phrases that describe the main topics of a document; that is, they describe concepts. Mihalcea and Csomai compute keyphraseness as property of n-grams rather than concepts. Furthermore, they compute it over the entire Wikipedia corpus: thus keyphraseness in their sense reflects the significance of a phrase for the document collection as a whole rather than for an individual document. For instance, the descriptor *Java (Programming language)* is more topical in a document that covers aspects of this language than in one that explains an algorithm that happens to be written in Java. Previously, to identify a document's topics, an analog of keyphraseness has been combined with document-specific features (Frank *et al*. 1999). We extend this to use the Wikipedia version of keyphraseness.

3. Indexing with Wikipedia as a controlled vocabulary

We follow the basic two-stage structure of most keyphrase indexing algorithms: first select many candidate keyphrases for a document and then filter out all but the most promising. In keyphrase *extraction* candidates are plain document phrases, while in keyphrase *indexing* they are descriptors from the controlled vocabulary. We use Wikipedia articles as candidates and their names as the topic descriptors. Figure 1 illustrates this: descriptors on the left map into articles on the right.

3.1 Selecting candidate articles

The *candidate selection* step extracts word n-grams from the document and matches them with terms in a controlled vocabulary. Current systems work within a particular domain and use a domain-specific vocabulary. Moving outside a specific domain by using a general controlled vocabulary presents significant difficulties. Wikipedia contains 2M terms (i.e. articles) and a further 2M synonyms (i.e. redirects). Almost every document phrase can be mapped to at least one article; most phrases map to several. It is essential for success to avoid unnecessary mappings by disambiguating the word senses.

We perform candidate selection in two stages:

- What words and phrases are important?
- Which Wikipedia articles do they relate to?

The first stage excludes words that contribute little to identifying the document's topics—that is, words that can be changed without affecting the topics expressed. We adapt the "keyphraseness" feature and choose as candidates all phrases for which this exceeds a predefined threshold. Earlier, Frank *et al.* (1999) computed an analogous metric from a manually indexed corpus—but it had to be large to cover all sensible domain terminology. With Wikipedia this feature is defined within the vocabulary itself.

The second stage links each candidate phrase to a Wikipedia article that captures its meaning. Of course, the ambiguity of language and its wealth of synonyms are both reflected in Wikipedia, so word-sense disambiguation is necessary. For example, the word *tree* in a document about depth-first search should be linked to the article *Tree (Data structure)* rather than to any biological tree.

To identify possible articles, Mihalcea and Csomai (2007) analyze link annotations in Wikipedia. If the candidate *bar* appears in links annotated as [[bar (law)|bar]] and [[bar (establishment)|bar]], the two Wikipedia articles *Bar (law)* and *Bar (establishment)* are possible targets. We achieved more accurate results by matching candidate phrases to articles (and redirects) rather than to anchor text.

If more than one article relates to a given n-gram, the next step is to disambiguate the n-gram's meaning. Mihalcea and Csomai investigate two approaches. Their *data-driven* method extracts local and topical features from the ambiguous n-gram, such as part-of-speech and context words, and computes the most probable mapping based on the distribution of these features in the training data. Their *knowledge-based* method computes the overlap of the paragraph in which the n-gram appears with the opening paragraph of the Wikipedia article. The first method is computationally challenging, requiring the entire Wikipedia corpus for training. The second performs significantly worse than a baseline that simply chooses the most likely mapping. Instead, we use a new *similarity-based* disambiguation technique based on similarity of possible articles to context articles computed for the same document.

We found that most documents contain at least a handful of n-grams that have only one related Wikipedia article. We use these as "context articles." For the remaining n-grams, we compute pairwise similarity between candidate articles and context articles. We combine the average similarity of a candidate article with the commonness of this mapping, given the n-gram, to compute the final mapping. The following subsection describes this process in detail.

3.2 Details of the candidate selection method

To identify important words and phrases in a document we first extract all word n-grams. For each n-gram *a*, we compute its probability of being a candidate (in other words, its keyphraseness) as follows:

$$Keyphraseness(a) \approx \frac{count(D_{Link})}{count(D_a)}$$

Here, $count(D_{Link})$ is the number of Wikipedia articles in which this n-gram appears as a link, and $count(D_a)$ is the total number of articles in which it appears.

The next step is to identify the article corresponding to each candidate. Wikipedia titles are preprocessed by case-folding and removing parenthetical text. Then the case-folded n-grams are matched to identify matching titles. If the match is to a redirect, the target article is retrieved. If it is to a disambiguation page, all articles listed as meanings in the first position of each explanation are collected.

The result is a set of possible article mappings. If it contains just one member, that is used as a context article. Once all such articles for a given document are collected, we use them to disambiguate phrases with more than one possible mapping. For this, we compute the average semantic similarity of each candidate article to all context articles identified for a given document (Milne and Witten, 2008). For each pair of articles *x* and *y* we retrieve the sets of hyperlinks *X* and *Y* that appear in the text of the articles, and compute their overlap $X \cap Y$. Given the total number *N* of articles in Wikipedia, the similarity of *x* and *y* is:

$$SIM_{x,y} = 1 - \frac{\max(\log|X|, \log|Y|) - \log|X \cap Y|}{N - \min(\log|X|, \log|Y|)}.$$

For each article in the set of possible mappings, we compute its average similarity to the context articles.

At this point it is necessary to take into account the overall popularity of the candidate articles as link targets. The *commonness* of article *T* being the target of a link with anchor text *a* is defined as

$$Commonness_{a,T} = \frac{P(a\,|\,T)}{P(a)}.$$

For example, the word *Jaguar* appears as a link anchor in Wikipedia 927 times. In 466 cases it links to the article *Jaguar cars*, thus the commonness of this mapping is 0.5. In 203 cases it links to the description of *Jaguar* as an animal, a commonness of 0.22. Mihalcea and Csomai (2007) use this information for one of their baselines, but seem to ignore it in the disambiguation process.

Finally, we multiply the article *T*'s average similarity to the context articles by its commonness given the n-gram *a*:

$$Score(a,T) = \frac{\sum_{c \in C} SIM_{T,c}}{|C|} \times Commonness_{a,T},$$

where $c \in C$ are the context articles for *T*. The highest-scoring article is chosen as the index term for the n-gram *a*.

3.3 Evaluation of candidate selection

To evaluate this disambiguation method we chose 100 random Wikipedia articles and used their manually annotated content as test documents. We iterate over the links in these articles, and use the above strategy to disambiguate them to Wikipedia articles. Table 1 compares the results with two baselines. The first chooses an article at random from the set of candidate mappings. The second chooses the article whose commonness value is greatest. The results demonstrate that the new similarity-based disambiguation method covers almost as many candidates as the baselines (17,416 vs. 17,640) and is significantly more accurate than both. Section 5.2 contains further evaluation of this technique based on multiple-indexer data.

3.3 Filtering

The *candidate selection* step is followed by a *filtering* step that characterizes each candidate article by statistical and semantic properties ("features") and determines the final score using a machine learning algorithm that calculates the importance of each feature from training data.

Earlier indexing schemes use features such as occurrence frequency and position in the document (Frank *et al.* 1999). However, results can be improved by including features contributed by Wikipedia itself—such as keyphraseness. Furthermore, it is known that performance

	A	C	P	R	F
Random	17,640	8,651	45.8	45.7	45.8
Most common	17,640	15,886	90.6	90.4	90.5
Similarity-based	17,416	16,220	93.3	92.3	92.9

Table 1. Disambiguation results: Attempted, Correct, Precision (%), Recall (%), F-measure (%)

improves significantly if account is taken of semantic relations between candidate phrases, as expressed in the controlled vocabulary (Medelyan and Witten, 2008). Although, strictly speaking, Wikipedia does not define semantic relations, articles can be seen as related if they contain many mutual hyperlinks (Milne and Witten, 2008). Alternatively, a similarity score can be computed based on their content (Gabrilovich and Markovich, 2007).

3.4 Features for learning

For any given document, the candidate selection stage yields a list of Wikipedia articles—terms—that describe the important concepts it mentions. Each term has a frequency that is the number of n-gram occurrences in the document that were mapped to it. Following Medelyan and Witten (2008), we define several features that indicate significance of a candidate term T in a document D.

1. **TF×IDF** = $\frac{\mathrm{freq}(T,D)}{\mathrm{size}(D)} \times -\log_2 \frac{\mathrm{count}(T)}{N}$,

This compares the frequency of a term in the document with its occurrence in general use. Here, freq(T,D) is term T's occurrence count in document D, size(D) is D's word count, and count(T) is the number of articles containing T in the training corpus.

2. **Position of first occurrence** of T in D, measured in words and normalized by D's word count. Phrases with extreme (high or low) values are more likely to be valid index terms because they appear either in the opening or closing parts of the document. Professional human indexers commonly focus on these portions in order to assign keyphrases to lengthy documents without having to read them completely (David *et al.*, 1995).

3. **Length** of T in words. Experiments have indicated that human indexers may prefer to assign multi-word terms.

4. **Node degree**, or how richly T is connected through thesaurus links to others that occur in the document. We define the degree of the Wikipedia article T as the number of hyperlinks that connect it to other articles in Wikipedia that have also been identified as candidate terms for the document. A document that describes a particular topic will cover many related concepts, so candidate articles with high node degree are more likely to be significant.

5. **Document-specific keyphraseness**. For each candidate article T we define *document-specific keyphraseness*$_{DS}$ to be the sum over keyphraseness values for all unique n-grams a that were mapped to this article, times their occurrence in the document *freq(a)*:

$$keyphraseness_{DS}(T) = \sum_{a \Rightarrow T} keyphraseness(a) \times freq(a)$$

3.5 Using the features to identify the index terms

Given these features, a model is built from training data—that is, documents to which terms have been manually assigned. For each training document, candidate terms are identified and their feature values calculated. Because our data is independently indexed by several humans, we assign a "degree of correctness" to each candidate. This is the number of human indexers who have chosen the term divided by the total number of indexers: thus a term chosen by 3 out of 6 indexers receives the value 0.5.

From the training data, the learning algorithm creates a model from that predicts the class from the feature values. We use the Naïve Bayes classifier in WEKA (Witten and Frank, 2005). To deal with non-standard distributions of the feature values, we apply John and Langley's (1995) kernel estimation procedure.

To identify topics for a new document, all its terms (i.e., candidate articles) and their feature values are determined. The model built during training is applied to determine the overall probability that each candidate is an index term, and those with the greatest probabilities are selected.

4. Evaluation

Topic indexing is usually evaluated by asking two or more human indexers to assign topics to the same set of test documents. The higher their consistency with each other, the greater the quality of indexing (Rolling, 1981). Of course, indexing is subjective and consistency is seldom high. To reliably evaluate an automatic scheme it should be compared against several indexers, not just one—the goal being to achieve the same consistency with the group as group members achieve with one another.

4.1 Experimental data

We chose 20 technical research reports covering different aspects of computer science. Fifteen teams of senior computer science undergraduates independently assigned topics to each report using Wikipedia article names as the allowed vocabulary. Each team had two members who worked together in two 1½ hour sessions, striving to achieve high indexing consistency with the other teams; no collaboration was allowed. Teams were instructed to assign around 5 terms to each document; on average they assigned 5.7 terms. Each document received 35.5 different terms, so the overlap between teams was low.

We analyzed the group's performance using a standard measure of inter-indexer consistency:

$$Consistency = \frac{2C}{A+B}$$

where A and B the total number of terms two indexers assign and C is the number they have in common (Rolling, 1981). This is equivalent to the F-Measure (Medelyan and Witten, 2008) and the Kappa statistic for indexing with very large vocabularies (Hripcsak and Rothschild, 2005).

Table 2 shows the consistency of each team with the other 14. It also indicates whether team members are native

Team ID	English?	Year	Consistency (%)
1	no	4.5	21.4
2	no	1	24.1
3	no	4	26.2
4	no	2.5	28.7
5	yes	4	30.2
6	mixed	4	30.8
7	yes	3	31.0
8	no	3	31.2
9	yes	4	31.6
10	yes	3.5	31.6
11	yes	4	31.6
12	mixed	3	32.4
13	yes	4	33.8
14	mixed	4	35.5
15	yes	4	37.1
		overall	**30.5**

Table 2. Consistency of each team with the others

English speakers, foreign students, or mixed, and gives the average study year of team members. Consistency ranges from 21.1% to 37.1% with an average of 30.5%. In a similar experiment professional indexers achieved an average consistency of 39% (Medelyan and Witten, 2008); however the controlled vocabulary was far smaller (28,000 vs. 2M concepts).

4.2 Results

We first evaluate the performance of candidate selection, a crucial step in the indexing process that involves both phrase selection and word sense disambiguation. How many of the Wikipedia articles that people chose for each document are identified as candidates?

Table 3 shows the coverage of all manually chosen articles (Recall R). It also shows those that were chosen by at least 3 humans (best Recall, Rb), which we view as more important. The rows compare two disambiguation techniques: a simple one that chooses the most common sense, and the similarity-based approach.

The results are shown for extracting n-grams with keyphraseness exceeding 0.01, which covers a reasonable number of manually assigned Wikipedia articles and provides a sufficient number of context articles. An average of 473 candidate articles are identified for each document. The *similarity-based* disambiguation algorithm locates 78% of the articles chosen by at least 3 human indexers, 4.3 percentage points better than the *most common* baseline. Improvement in total recall is only 2.5 points, which indicates that the articles chosen by more indexers are more ambiguous, for example: *Tree (data structure)*, *Inheritance (compute science)*, *Index (search engine)*.

Table 4 evaluates the filtering technique of Section 3.3–3.6 by comparing its performance with the Wikipedia articles assigned by 15 human teams. As a baseline we extract for each document the 5 Wikipedia articles with the highest TF×IDF values (row 2). This achieves an average consistency with humans of 17.5%. Next we evaluate the filtering strategy based on features previously used for automatic indexing: features 1–4 of Section 3.4 (row 3). We use "leave-one-out" evaluation, i.e. train on 19 documents and test on the remaining one, until all documents have been indexed. The average result, 25.5%, is 8 points above the TF×IDF baseline.

	# articles per doc	P	R	Rb
most common	388	5.1	52.5	73.8
similarity-based	473	5.6	55.0	78.1

Table 3. Candidate selection results: Precision, Recall, best Recall (Rb) (%)

		Consistency (%)		
	Method	**min**	**avg**	**max**
1	human indexers	20.3	**30.5**	**38.4**
2	TF×IDF baseline	10.9	17.5	23.5
3	ML with 4 features	20.0	25.5	29.6
4	keyphraseness$_{DS}$	22.5	27.5	32.1
5	ML with 5 features	**24.5**	**30.5**	36.1

Table 4. Performance compared to human indexers

Now we evaluate the use of the document-specific keyphraseness feature *keyphraseness*$_{DS}$ (feature 5 of Section 3.4) (row 4). The consistency of the top 5 candidate articles is 27.5%, only 3 points less than consistency among humans. Finally we combine *keyphraseness*$_{DS}$ with the other 4 features, bringing the average consistency to 30.5% (row 5). This is the same as the average over the 15 human teams (Table 2). The new method outperforms 5 teams, all in their 4th year of study in the same area as the test documents; one team consists of two English native speakers. These results are achieved after learning from only 19 manually indexed documents.

4.3 Examples

Figure 2 illustrates the terms assigned by humans (open circles) and our algorithm (filled circles). The 6 best human teams are shown in different colors; other teams are in black. Arrows between nodes show hyperlinks in the corresponding Wikipedia articles, and indicate the semantic relatedness of these concepts. The behavior of the algorithm is indistinguishable from that of the student teams.[1]

5. Conclusions

This paper combines research on linking textual documents into Wikipedia (Mihalcea and Csomai, 2007) with research on domain-specific topic indexing (Medelyan and Witten, 2008). We treat Wikipedia articles as topics and their titles as controlled terms, or descriptors.

We first link all important phrases in a document to Wikipedia articles by matching them to titles of articles, redirects and disambiguation pages. When multiple mappings exist, we apply an unsupervised disambiguation

[1] See *http://www.cs.waikato.ac.nz/~olena/wikipedia.html* for full results.

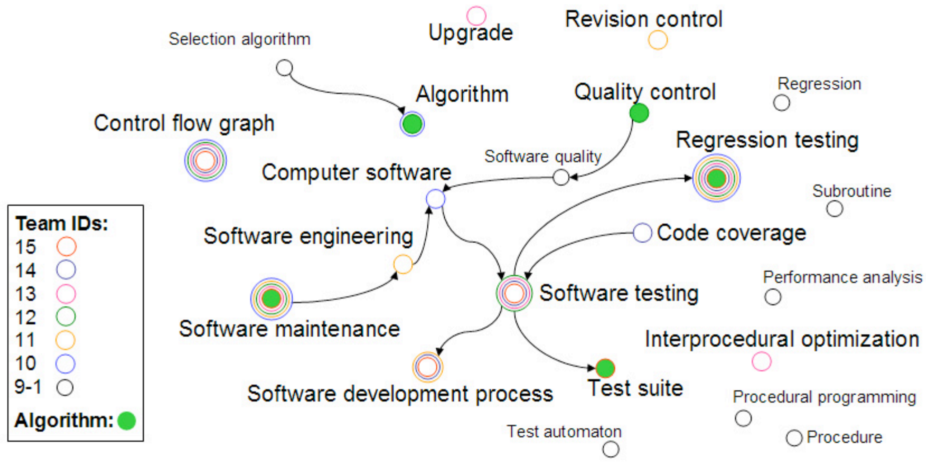

Figure 2. Topics assigned to a document entitled "A Safe, Efficient Regression Test Selection Technique" by human teams (outlined circles) and the new algorithm (filled circles)

procedure based on semantic similarity. This new method outperforms the unsupervised disambiguation proposed earlier, and achieves an F-measure of 93%.

Next, we restrict all linked Wikipedia articles to a handful of significant ones representing the document's main topics. One technique utilizes the knowledge in Wikipedia; a second uses training data to learn the distribution of properties typical for manually assigned topics. Evaluation on computer science reports indexed by human indexers shows that the former technique outperforms the latter, and a combination of the two yields the best results. The final approach has the same consistency with the 15 human teams as their average consistency with themselves.

Note that this performance is achieved with a very small training set of 19 documents, with 15 keyphrase sets each. Our new algorithm for efficient indexing with Wikipedia can assign topics to documents in nearly any domain and language, and we plan to capitalize on this by applying it to the multiply-indexed documents on social bookmarking sites like *del.icio.us* and *citeulike.org*.

Acknowledgements

We thank the students in our department who participated in the indexing competition, Andras Csomai for providing his data, and Eibe Frank for valuable comments.

References

Aronson, A. R.; Bodenreider, O.; Chang, H. F. et al. 2000. The NLM indexing initiative. *Proc. Fall Symp. of the American Medical Informatics Association*, LA, pp. 17–21.

David, C.; Giroux, L.; Bertrand-Gastaldy, S.; and Lanteigne, D. 1995. Indexing as problem solving. *Proc. ASIS Annual Meeting*, v32. pp. 49–55.

Frank, E.; Paynter, G.W.; Witten, I.H.; Gutwin, C.; and Nevill-Manning, C.G. 1999. Domain-specific keyphrase extraction. *Proc. Int. Joint Conference on Artificial Intelligence*, Stockholm, Sweden, pp. 668–673.

Gabrilovich, E.; and Markovitch, S. 2006. Overcoming the brittleness bottleneck using Wikipedia. *Proc. National Conference on Artificial Intelligence*, Boston, MA.

Gay, C.W.; Kayaalp, M.; and Aronson, A. R. 2005. Semi-automatic indexing of full text biomedical articles. *Proc Fall Symp. of the American Medical Informatics Association*, Washington DC, USA, pp. 271–275.

Hripcsak, G.; and Rothschild, A.S. 2005. Agreement, the F-Measure, and reliability in information retrieval. *J. Am. Medical Information Association*, 12(3): 296–298.

Markó, K.; Hahn, U.; Schulz, S.; Daumke, P.; and Nohama, P. 2004. Interlingual indexing across different languages. *Proc. Int. Conf. Recherche d'Information Assistée par Ordinateur*, pp. 100–115.

Medelyan, O.; and Witten, I.H. 2008. Domain independent automatic keyphrase indexing with small training sets. To appear in *J. Am. Soc. Information Science and Technology*.

Mihalcea, R.; and Csomai, A. 2007. Wikify!: linking documents to encyclopedic knowledge. *Proc. CIKM*, pp. 233-242.

Milne, D.; Medelyan, O.; and Witten, I.H. 2006. Mining domain-specific thesauri from Wikipedia: A case study. *Proc. IEEE Int. Conf. on Web Intelligence*, Hong Kong.

Milne, D.; and Witten, I.H. 2008. An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. Forthcoming.

Rolling, L. 1981. Indexing consistency, quality and efficiency. *Info. Processing and Management*, 17 (2): 69–76.

Witten, I.H.; and Frank, E. 1999. *Data mining: Practical machine learning tools and techniques with Java implementations*. Morgan Kaufmann, San Francisco, CA.