Learning to Extract Folktale Keywords

Dolf Trieschnigg, Dong Nguyen and Mariët Theune
University of Twente
Enschede, The Netherlands
{d.trieschnigg,d.nguyen,m.theune}@utwente.nl

Abstract

Manually assigned keywords provide a valuable means for accessing large document collections. They can serve as a shallow document summary and enable more efficient retrieval and aggregation of information. In this paper we investigate keywords in the context of the Dutch Folktale Database, a large collection of stories including fairy tales, jokes and urban legends. We carry out a quantitative and qualitative analysis of the keywords in the collection. Up to 80% of the assigned keywords (or a minor variation) appear in the text itself. Human annotators show moderate to substantial agreement in their judgment of keywords. Finally, we evaluate a learning to rank approach to extract and rank keyword candidates. We conclude that this is a promising approach to automate this time intensive task.

1 Introduction

Keywords are frequently used as a simple way to provide descriptive metadata about collections of documents. A set of keywords can concisely present the most important aspects of a document and enable quick summaries of multiple documents. The word cloud in Figure 1, for instance, gives a quick impression of the most important topics in a collection of over 40,000 documents (a collection of Dutch folktales).

Keyword assignment or generation is the task of finding the most important, topical keywords or keyphrases to describe a document (Turney, 2000; Frank et al., 1999). Based on keywords, small groups of documents (Hammouda et al., 2005) or large collections of documents (Park et al., 2002) can be summarized. Keyword *extraction* is a restricted case of keyword assignment: the assigned keywords are a selection of the words or phrases appearing in the document itself (Turney, 2000; Frank et al., 1999).

In this paper we look into keyword extraction in the domain of cultural heritage, in particular for extracting keywords from folktale narratives found in the Dutch Folktale Database (more on this collection in section 3). These narratives might require a different approach for extraction than in other domains, such as news stories and scholarly articles (Jiang et al., 2009). Stories in the Dutch Folktale Database are annotated with uncontrolled, free-text, keywords. Because suggesting keywords which do not appear in the text is a considerably harder task to automate and to evaluate, we restrict ourselves to keywords extracted from the text itself.

In the first part of this paper we study the current practice of keyword assignment for this collection. We analyze the assigned keywords in the collection as a whole and present a more fine-grained analysis of a sample of documents. Moreover, we investigate to what extent human annotators agree on suitable keywords extracted from the text. Manually assigning keywords is an expensive and time-consuming process. Automatic assignment would bring down the cost and time to archive material. In the second part of this paper we evaluate a number of automatic keyword extraction methods. We show that a learning to rank approach gives promising results.

The overview of this paper is as follows. We first describe related work in automatic keyword assignment. In section 3 we introduce the Dutch Folktale Database. In section 4 we present an analysis of the keywords currently used in the folktale database. In section 5 we investigate the agreement of human annotators on keyword extraction. In section 6 we present and evaluate an automatic method for extracting and ranking keywords. We end with a discussion and conclusion in section 7.

Proceedings of the 7th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities, pages 65–73,
Sofia, Bulgaria, August 8 2013.

Keyword (translation)	Frequency
dood (death)	5,861
man (man)	4,547
vrouw (woman)	4,154
sterven (to die)	3,168
huis (house)	2,894
spokerij (haunting)	2,491
duivel (devil)	2,487
nacht (night)	2,449
voorteken (omen)	2,380
voorloop (forerunnings)	2,372
geld (money)	2,370
toverij (sorcery)	2,322
zien (to see)	2,270
heks (witch)	2,233
boer (farmer)	2,189
water (water)	2,177
angst (fear)	2,091
hekserij (witchcraft)	1,911
kind (child)	1,853
spoken (ghosts)	1,747
spook (ghost)	1,742
seks (sex)	1,659

Figure 1: Frequent keywords in the Dutch Folktale Database

2 Related Work

Because of space limitations, we limit our discussion of related work to keyword extraction in the context of free-text indexing. Automated *controlled* vocabulary indexing is a fundamentally different task (see for instance Medelyan and Witten (2006) and Plaunt and Norgard (1998)).

Typically, keyword extraction consists of two steps. In the first step candidate keywords are determined and features, such as the frequency or position in the document, are calculated to characterize these keywords. In the second step the candidates are filtered and ranked based on these features. Both unsupervised and supervised algorithms have been used to do this.

2.1 Candidate Extraction

Candidate keywords can be extracted in a number of ways. The simplest approach is to treat each single word as a candidate keyword, optionally filtering out stop words or only selecting words with a particular Part-of-Speech (Liu et al., 2009a; Jiang et al., 2009). More sophisticated approaches allow for multi-word keywords, by extracting consecutive words from the text, optionally limited to keywords adhering to specific lexical patterns (Osinski and Weiss, 2005; Hulth, 2003; Rose et al., 2010; Frank et al., 1999; Turney, 2000).

2.2 Features to Characterize Keywords

Many features for characterizing candidate keywords have been investigated previously, with varying computational complexities and resource requirements. The simplest features are based on document and collection statistics, for instance the frequency of a potential keyword in the document and the inverse document frequency in the collection (Turney, 2000; Hulth, 2003; Frank et al., 1999). Examples of more complex features are: features based on characteristics of lexical chains, requiring a lexical database with word meanings (Ercan and Cicekli, 2007); features related to frequencies in external document collections and query logs (Bendersky and Croft, 2008; Yih et al., 2006; Liu et al., 2009b; Xu et al., 2010); and a feature to determine the cohesiveness of retrieved documents with that keyword (Bendersky and Croft, 2008).

2.3 Unsupervised Methods for Keyword Extraction

Unsupervised methods for keyword extraction typically rely on heuristics to filter and rank the keywords in order of importance. For instance, by ranking the candidates by their importance in the collection – estimated by the inverse document frequency. Another approach is to apply the PageRank algorithm to determine the most important keywords based on their co-occurrence link-structure (Mihalcea and Tarau, 2004). Liu et al. (2009b) employed clustering to extract keywords that cover *all* important topics from the original text. From each topic cluster an exemplar is determined and for each exemplar the best corresponding keyword is determined.

2.4 Supervised Methods for Keyword Extraction

Early supervised methods used training data to set the optimal parameters for (unsupervised) systems

based on heuristics (Turney, 2000). Other methods approached keyword extraction as a binary classification problem: given a candidate keyword it has to be classified as either a keyword or not. Methods include decision trees (Bendersky and Croft, 2008), Naive Bayes (Frank et al., 1999) and Support Vector Machines (Zhang et al., 2006). Zhang et al. (2008) approached keyword extraction as a labeling problem for which they employed conditional random fields. Recently, keyword extraction has been cast as a ranking problem and learning to rank techniques have been applied to solve it (Jiang et al., 2009). Jiang et al. (2009) concluded that learning to rank approaches performed better than binary classifiers in the context of extracting keywords from scholarly texts and websites. Different variations of learning to rank exist, see (Li, 2011) for an overview.

3 The Dutch Folktale Database

The Dutch Folktale Database is a repository of over 40,000 folktales in Dutch, old Dutch, Frisian and a large number of Dutch dialects. The material has been collected in the 19th, 20th and 21th centuries, and consists of stories from various periods, including the Middle Ages and the Renaissance. The collection has both an archival and a research function. It preserves an important part of the oral cultural heritage of the Netherlands and can be used for comparative folk narrative studies. Since 2004 the database is available online[1].

The real value of the database does not only lie the stories themselves, but also in their manually added set of descriptive metadata fields. These fields include, for example, a summary in Dutch, a list of proper names present in the folktales, and a list of keywords. Adding these metadata is a time-consuming and demanding task. In fact, the amount of work involved hampers the growth of the folktale database. A large backlog of digitized folktales is awaiting metadata assignment before they can be archived in the collection. Being able to automatically assign keywords to these documents would be a first step to speed up the archiving process.

4 Analysis of Assigned Keywords

In this section we analyze the keywords that have been manually assigned to the stories in the Dutch Folktale Database. First we look at the keywords assigned to the collection as a whole. After that we make a more fine-grained analysis of the keywords assigned to a selection of the documents.

[1] `http://www.verhalenbank.nl`, in Dutch only

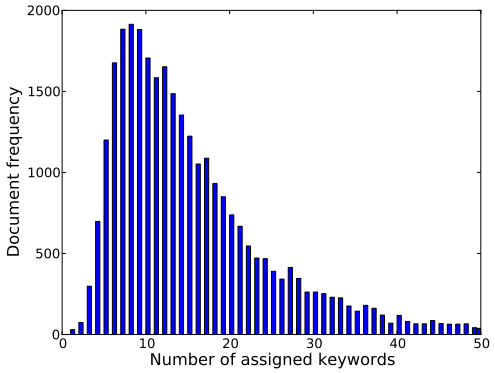

Figure 2: Number of assigned keywords per document

4.1 Quantitative Analysis

We analyzed a snapshot from the Dutch Folktale Database (from early 2012) that consists of 41,336 folktales. On average, 15 keywords have been assigned to each of these documents (see Figure 2). The median number of assigned keywords is 10, however. The keywords vocabulary has 43,195 unique keywords, most of which consist of a single word (90%). Figure 1 shows a word cloud of keywords used in the collection; more frequent keyword types appear larger. On the right, it lists the most frequent keyword types (and their translations). The assignment of keywords to documents has a Zipfian distribution: a few keyword types are assigned to many documents, whereas many keyword types are assigned to few documents.

When we limit our collection to stories in Dutch (15,147 documents), we can determine how many of the manually assigned keywords can be found literally in the story text[2]. We define the *keyword coverage* of a document as the fraction of its assigned keywords which is found in the full text or its summary. The average keyword coverage of the Dutch stories is 65%. Figure 3 shows a histogram of the coverage. It shows that most of the documents have a keyword coverage of 0.5 or more.

[2] Stories in other languages or dialects have been assigned Dutch keywords.

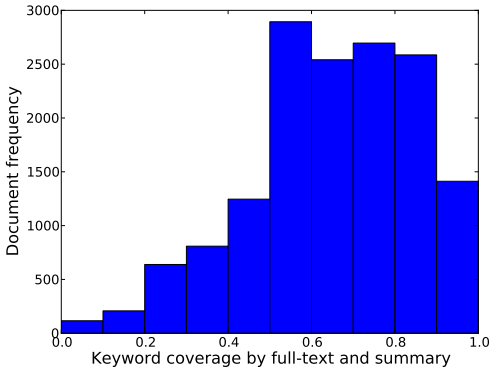

Figure 3: Keyword coverage of folktales in Dutch

4.2 Qualitative Analysis

The quantitative analysis does not provide insight into what kind of keywords have been assigned. Therefore, we analyzed a selection of documents more thoroughly. For each of the five largest genres in the collection (fairy tale, traditional legend, joke, urban legend and riddle) we sampled 10 tales and manually classified the keywords assigned to these folktales. A total of almost 1000 keywords was analyzed. Table 1 summarizes the statistics of this analysis. Almost 80% of the keywords appear literally or almost literally in the text. The almost literal appearances include keywords which differ in quantity (plural versus singular form) and verb forms. Verb forms vary in tense (present rather than past tense) and infinitive keywords of separable verbs. An example of the latter is the assignment of the keyword "terugkeren", to return, where "keren" ($\sim$ turn) and "terug" ($\sim$ back) are used in a sentence. Of the analyzed keywords 5% are synonyms of words appearing the text and 2.3% are hypernyms of words appearing the text (e.g. "wapen", weapon, is used as a keyword with "mes", knife, mentioned in the text). The remaining 13% of the keywords represent abstract topic, event and activity descriptions. For example, the keyword "wegsturen", to send away, when one of the characters explicitly asks someone to leave. Other examples are the keywords "baan", job, and "arbeid", labor, when the story is about an unemployed person.

Based on these numbers we can conclude that based on extraction techniques alone we should be able to reproduce a large portion of the manual keyword assignment. When thesauri are employed to find synonyms and hypernyms, up to 87% of the manually assigned keywords could be found. A much harder task is to obtain the remaining 13% of more abstract keywords, which we will study in future research.

Classification	Count	Perc.
Literal	669	67.6%
Almost literal	120	12.1%
Synonym	49	5.0%
Hypernym	23	2.3%
Typing error	2	0.2%
Other	126	12.7%
Total	989	100.0%

Table 1: Keyword types in a set of 1000 folktales

5 Evaluating Agreement in Keyword Assignment

The previous analyses raise the question whether the keywords have been consistently assigned: do annotators choose the same keywords when presented with the same text? Moreover, knowing the difficulty of the task for human annotators will give us an indication of the level of performance we may expect from automatic keyword assignment. To determine the agreement between annotators we asked ten annotators to classify the vocabulary of five folktales from different genres. Frog[3] (van den Bosch et al., 2007) was used to extract the vocabulary of lemmas. After carefully reading a folktale, the annotator classified the alphabetically sorted list of lemmas extracted from the text. Each lemma was classified as either: 1) not a relevant keyword – should not be assigned to this document (*non*); 2) a relevant keyword – should be assigned (*rel*); 3) a *highly* relevant keyword – should definitely be assigned (*hrel*). The three levels of relevance were used to see whether annotators have a preference for certain keywords. The pairwise agreement between annotators was measured using Cohen's kappa. Each document was judged twice, totaling a set of 25 documents. Most of the annotators were familiar with the folktale database and its keywords; two were active contributors to the database and thus had previous experience in assigning keywords to folktales.

On average, the annotators judged 79% of the vocabulary as non-relevant as keywords. 9% and 12% of the vocabulary was judged as relevant and highly relevant respectively, but there was a large variation in these percentages: some annotators assigned more highly relevant keywords, others assigned more relevant keywords.

[3] http://ilk.uvt.nl/frog/

Classes	Cohen's Kappa Average	σ	Min	Max
non, rel, hrel	0.48	0.14	0.16	0.77
non, rel + hrel	0.62	0.16	0.25	0.92
non + rel, hrel	0.47	0.20	0.0	0.84

Table 2: Classification agreement between annotators. Non: non-relevant, rel: relevant, hrel: highly relevant.

The two experienced annotators showed a consistently higher average agreement in comparison to the other annotators (0.56 and 0.50 for non, rel, hrel; 0.7 and 0.64 for non, rel + hrel; 0.56 and 0.50 for non + rel, hrel). Moreover, they assigned more (relevant and highly relevant) keywords to the documents on average.

Table 2 summarizes the agreement measured between annotators. The first row indicates the agreement when considering agreement over all three classes; the second row indicates the agreement when treating relevant and highly relevant keywords as the same class; the last row shows the agreement in indicating the same highly relevant keywords. The numbers indicate moderate agreement between annotators over all three classes and when considering the choice of highly relevant keywords. Annotators show substantial agreement on deciding between non-relevant and relevant keywords. Table 3 shows the agreement between annotators on keywords with different parts of speech (CGN[4] tagset). Most disagreements are on nouns, adjectives and verbs. Verbs and adjectives show few agreements on relevant and highly relevant keywords. In contrast, on 20% of the nouns annotators agree on their relevance. It appears that the annotators do not agree whether adjectives and verbs should be used as keywords at all. We can give three other reasons why annotators did not agree. First, for longer stories annotators were presented with long lists of candidate keywords. Sometimes relevant keywords might have been simply overlooked. Second, it turned out that some annotators selected some keywords in favor to other keywords (for instance a hyponym rather than a hypernym), where others simply annotated both as relevant. Third, the disagreement can be explained by lack of detailed instructions. The annotators were not told how many (highly) relevant keywords to select or what criteria should be met by the keywords. Such instructions are not available to current annotators of the collection either.

We conclude that annotators typically agree on the keywords from a text, but have a varying notion of highly relevant keywords. The average keywords-based representation strongly condenses the documents vocabulary: a document can be represented by a fifth (21%) of its vocabulary[5]. This value can be used as a cut-off point for methods ranking extracted keywords, discussed hereafter.

6 Automatically Extracting Keywords

In the last part of this paper we look into automatically extracting keywords. We compare a learning to rank classifier to baselines based on frequency and reuse in their ability to reproduce keywords found in manually classified folktales.

In all cases we use the same method for extracting keyword candidates. Since most of the manual keywords are single words (90% of the used keyword types in the collection), we simply extract single words as keyword candidates. We use Frog for tokenization and part of speech tagging. Stop words are not removed.

6.1 Baseline Systems

We use a basic unsupervised baseline for keyword extraction: the words are ranked according to descending TF-IDF. We refer to this system as *TF-IDF*. TF, *term frequency*, and IDF, *inverse document frequency*, are indicators of the term's local and global importance and are frequently used in information retrieval to indicate the relative importance of a word (Baeza-Yates and Ribeiro-Neto, 2011).

Note that a word appearing once in the collection has the highest IDF score. This would imply that the most uncommon words are also the most important resulting in a bias towards spelling errors, proper names, and other uncommon words. Hence, our second baseline takes into account whether a keyword has been used before in a training set. Again, the candidates are ranked by descending TF-IDF, but now keywords appearing in the training collection are ranked above the keywords not appearing in the collection. We refer to this baseline as *TF-IDF-T*.

[4]Corpus Gesproken Nederlands (Spoken Dutch Corpus), `http://lands.let.kun.nl/cgn/ehome.htm`

[5]Based on the figures that on average 9% of the vocabulary is judged as relevant and 12% as highly relevant

	Part of speech	Adjective	Adverb	Noun	Special	Numeral	Prep.	Verb
	Number of words	272	257	646	131	53	268	664
Agreement	non	70%	96%	40%	95%	81%	99%	73%
	rel	4%	0%	6%	0%	0%	0%	3%
	hrel	1%	0%	14%	2%	2%	0%	4%
Disagreement	non ↔ rel	15%	2%	17%	2%	11%	0%	12%
	non ↔ hrel	5%	1%	8%	2%	4%	1%	5%
	rel ↔ hrel	5%	0%	15%	0%	2%	0%	4%

Table 3: Agreement and disagreement of annotators on keywords with different parts of speech. Values are column-wise percentages. Tags with full agreement are not shown.

6.2 Learning to Rank Keywords

Following Jiang et al. (2009) we apply a learning to rank technique to rank the list of extracted keywords. We train an SVM to classify the relative ordering of pairs of keywords. Words corresponding to manual keywords should be ranked higher than other words appearing in the document. We use SVM-rank to train a linear ranking SVM (Joachims, 2006). We use the following features.

6.2.1 Word Context

We use the following word context features:

starts uppercase: indicates whether the token starts with an uppercase letter (1) or not (0). Since proper names are not used as keywords in the folktale database, this feature is expected to be a negative indicator of a word being a keyword.

contains space: indicates whether the token contains a space (Frog extracts some Dutch multiword phrases as a single token). Tokens with spaces are not very common.

is number: indicates whether the token consists of only digits. Numbers are expected not to be a keyword.

contains letters: indicates whether the token contains at least a single letter. Keywords are expected to contain letters.

all capital letters: indicates whether the token consists of only capital letters. Words with only capital letters are not expected to be keywords.

single letter: indicates whether the token consists of only one letter. One letter keywords are very uncommon.

contains punctuation: indicates whether the token contains punctuation such as apostrophes. Keywords are expected not to contain punctuation.

part of speech: indicates the part of speech of the token (each tag is a binary feature). Nouns are expected to be a positive indicator of keywords (Jiang et al., 2009).

6.2.2 Document Context

We use the following document context features:

tf: the term frequency indicates the number of appearances of the word divided by the total number of tokens in the document.

first offset: indicates the offset of the word's first appearance in the document, normalized by the number of tokens in the document (following Zhang et al. (2008)). Important (key)words are expected to be mentioned early.

first sentence offset: indicates the offset of the first sentence in which the token appears, normalized by the number of sentences in the document.

sentence importance: indicates the maximum importance of a sentence in which the word appears, as measured by the SumBasic score (Nenkova and Vanderwende, 2005). SumBasic determines the relative importance of sentences solely on word probability distributions in the text.

dispersion: indicates the dispersion or scattering of the word in the document. Words which are highly dispersed are expected to be more important. The DP_{norm} is used as a dispersion measure, proposed in Gries (2008).

6.2.3 Collection Context

We use the following features from the collection/training context:

idf: the inverse document frequency indicates the collection importance of the word based on frequency: frequent terms in the collection are less important than rare terms in the collection.

tf.idf: combines the *tf* and idf features by multiplying them. It indicates a trade-off between local and global word importance.

is training keyword: indicates whether the word is used in the training collection as a keyword.

assignment ratio: indicates the percentage of documents in which the term is present in the text and in which it is also assigned as a keyword.

6.3 Evaluation Method

We evaluate the ranking methods on their ability to reproduce the manual assignment of keywords. Ideally the ranking methods rank these manual keywords highest. We measure the effectiveness of ranking in terms of (mean) average precision (MAP), precision at rank 5 (P@5) and precision at rank R (P@R), similar to Jiang et al. (2009). Note that we use *all* the manually assigned keywords as a ground truth, including words which do not occur in the text itself. This lowers the highest achievable performance, but it will give a better idea of the performance for the real task.

We perform a 10-fold stratified cross-validation with a set of 10,900 documents from the Dutch Folktale Database, all written in modern Dutch.

6.4 Results

Table 4 lists the performance of the three tested systems. The *TF-IDF* system performs worst, and is significantly outperformed by the *TF-IDF-T* system, which in turn is significantly outperformed by the *rank-SVM* system. On average, rank-SVM returns 3 relevant keywords in its top 5. The reported mean average precision values are affected by manual keywords which are not present in the text itself. To put these numbers in perspective: if we would put the manual keywords which are in the text in an optimal ranking, i.e. return these keywords first, we would achieve an upper bound mean average precision of 0.5675. Taking into account the likelihood that some of the highly ranked false positives are relevant after all (the annotator might have missed a relevant keyword) and considering the difficulty of the task (given the variation in agreement between manual annotators), we argue that the rank-SVM performs quite well.

Jiang et al. (2009) reported MAPs of 0.288 and 0.503 on the ranking of extracted keyphrases from scholarly articles and tags from websites respectively. Based on these numbers, we could argue that assigning keywords to folktales is harder than reproducing the tags of websites, and slightly easier than reproducing keyphrases from scientific articles. Because of differences in the experimental setup (e.g. size of the training set, features and system used), it is difficult to make strong claims on the difficulty of the task.

System	MAP	P@5	P@R
TF-IDF	0.260	0.394	0.317
TF-IDF-T	0.336	0.541	0.384
rank-SVM	**0.399**	**0.631**	**0.453**

Table 4: Keyword extraction effectiveness. The differences between systems are statistically significant (paired t-test, $p < 0.001$)

	Change in		
Feature	MAP	P@5	P@R
assignment ratio	-0.036	-0.056	-0.038
is training keyword	0.006	0.002	0.005
tf.idf	-0.004	-0.010	-0.002
part of speech	-0.003	-0.007	0.000
dispersion	-0.001	-0.001	0.000
idf	0.001	0.002	0.000
starts uppercase	0.000	0.000	-0.001
first offset	0.000	0.000	0.000
tf	0.000	0.000	0.000
contains space	0.000	0.000	0.000
is number	0.000	0.000	0.000
all capital letters	0.000	0.000	0.000
contains punctuation	0.000	0.000	0.000
contains letters	0.000	0.000	0.000
sentence importance	0.000	0.000	0.000
first sentence offset	0.000	0.000	0.000
single letter	0.000	0.000	0.000

Table 5: Differences in performance when leaving out features. The features are ordered by descending difference in MAP.

6.5 Feature Ablation

To determine the added value of the individual features we carried out an ablation study. Table 5 lists the changes in performance when leaving out a particular feature (or group of features in case of part of speech). It turns out that many features can be left out without hurting the performance. All the features testing simple word characteristics (such as single letter) do not, or only marginally influence the results. Also taking into account the importance of sentences (sentence importance), or the first appearance of a word (first offset and first sentence offset) does not contribute to the results.

System	MAP	P@5	P@R
rank-SVM	0.399	0.631	0.453
minimum set	**0.405**	**0.631**	**0.459**

Table 6: Results using the full set of features and the minimum set of features (assignment ratio, tf.idf, part of speech and dispersion). Differences between systems are statistically significant (t-test, $p < 0.001$).

Genre (# stories)	MAP	P@5	P@R
Trad. legend (3783)	**0.439**	**0.662**	**0.494**
Joke (2793)	**0.353**	**0.599**	**0.405**
Urban legend (1729)	0.398	**0.653**	0.459
Riddle (1067)	0.391	**0.573**	**0.415**
Fairy tale (558)	0.404	**0.670**	**0.477**
Pers. narrative (514)	**0.376**	**0.593**	0.437
Legend (221)	0.409	0.622	0.478
None (122)	0.366	0.602	0.421
Other (113)	0.405	0.648	0.472
All (10900)	0.399	0.631	0.453

Table 7: SVM performance split according to story genre. Values in bold are significantly different from the results on the other genres (independent t-test, p-value < 0.01)

These observations suggest that almost identical results can be obtained using only the features assignment ratio, tf.idf, part of speech and dispersion. The results reported in Table 6 confirm this (we do note that these results were obtained by optimizing on the test set).

6.6 Performance on Folktale Genres

The folktale database contains stories from different folktale genres, varying from legends to fairy tales and jokes. Table 7 lists the performance measures per story genre. Values in bold indicate significant differences with the stories from the other genres combined. The performance on traditional legends turns out to be significantly better than other genres: this could be explained by the fact that on average these stories are longer and therefore contain more keywords. Similarly, the decrease can be explained for jokes, which are much shorter on average. Another explanation could be that more abstract keywords are used to indicate the type of joke. Interestingly, the riddles, which are even shorter than jokes, do not perform significantly worse than the other genres. Personal narratives also underperformed in comparison to the other genres. We cannot readily explain this, but we suspect it may have something to do with the fact that personal narratives are more varied in content and contain more proper names.

7 Discussion and Conclusion

In this work we analyzed keywords in the context of the Dutch Folktale Database. In this database, on average 15 keywords have been assigned to a story, many of which are single keywords which appear literally or almost literally in the text itself. Keyword annotators show moderate to substantial agreement in extracting the same keywords for a story. We showed that a learning to rank method using features based on assignment ratio, tf.idf, part of speech and dispersion can be effectively used to extract and rank keyword candidates. We believe that this system can be used to suggest highly relevant keyword candidates to human annotators to speed up the archiving process.

In our evaluation we aimed to reproduce the manual annotations, but it is unclear whether better performing systems are actually more helpful to the user. In an ad hoc retrieval scenario, in which the user issues a single query and reviews a list of retrieved documents, extracted keywords might be used to boost the early precision of the results. However, a user might not even notice a difference when a different keyword extraction system is used. Moreover, the more abstract keywords which do not appear in the text might be more important for the user experience. In future work we want to get insight in how keywords contribute to the end user experience. Ideally, the evaluation should directly measure how useful the various keywords are for accessing the collection.

In this work we considered only *extracting* keywords from the text we want to annotate. Given the multilingual content of the database this is a limited approach: if the goal of assigning keywords is to obtain a normalized representation of the stories, this approach will require translation of either the source text (before extraction) or the extracted keywords. Even in the monolingual scenario, the extraction of keywords is limited in dealing with differences in style and word use. Writers may use different words or use words in a different way; ideally the representation based on keywords is a normalized representation which closes this semantic gap. In future work we will look into annotation with keywords from multi-lingual thesauri combined with free-text keywords extracted from the text itself. Finally, we want to look into classification of abstract themes and topics.

Acknowledgments

This research was supported by the Folktales as Classifiable Texts (FACT) project, part of the CATCH programme funded by the Netherlands Organisation for Scientific Research (NWO).

References

R Baeza-Yates and B. Ribeiro-Neto. 2011. *Modern Information Retrieval*. The Concepts and Technology Behind Search. Addison-Wesley.

M. Bendersky and W.B. Croft. 2008. Discovering key concepts in verbose queries. In *Proceedings of SIGIR 2008*, pages 491–498.

G. Ercan and I. Cicekli. 2007. Using lexical chains for keyword extraction. *Information Processing & Management*, 43(6):1705–1714.

E. Frank, G.W. Paynter, I.H. Witten, C. Gutwin, and C.G. Nevill-Manning. 1999. Domain-specific keyphrase extraction. In *Proceedings of IJCAI-99*, pages 668—673. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Stefan Th. Gries. 2008. Dispersions and adjusted frequencies in corpora. *International Journal of Corpus Linguistics*, 13(4):403–437.

K. Hammouda, D. Matute, and M. Kamel. 2005. Corephrase: Keyphrase extraction for document clustering. *Machine Learning and Data Mining in Pattern Recognition*, pages 265–274.

A. Hulth. 2003. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of EMNLP*, volume 10, pages 216–223, Morristown, NJ, USA. Association for Computational Linguistics.

X. Jiang, Y. Hu, and H. Li. 2009. A ranking approach to keyphrase extraction. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 756–757. ACM.

T. Joachims. 2006. Training Linear SVMs in Linear Time. In *the 12th ACM SIGKDD international conference*, pages 217–226, New York, NY, USA. ACM.

H. Li. 2011. *Learning to Rank for Information Retrieval and Natural Language Processing*. Synthesis Lectures on Human Language Technology. Morgan & Claypool Publishers.

F. Liu, D. Pennell, F. Liu, and Y. Liu. 2009a. Unsupervised approaches for automatic keyword extraction using meeting transcripts. In *Proceedings of NAACL 2009*, pages 620–628. Association for Computational Linguistics.

Z. Liu, P. Li, Y. Zheng, and M. Sun. 2009b. Clustering to find exemplar terms for keyphrase extraction. In *Proceedings of EMNLP*, pages 257–266. Association for Computational Linguistics.

O Medelyan and Ian H Witten. 2006. Thesaurus based automatic keyphrase indexing. In *JCDL 2006*, pages 296–297. ACM.

R. Mihalcea and P. Tarau. 2004. Textrank: Bringing order into texts. In *Proceedings of EMNLP*, volume 4, pages 404–411. Barcelona, Spain.

A. Nenkova and L. Vanderwende. 2005. The impact of frequency on summarization. *Microsoft Research, Redmond, Washington, Tech. Rep. MSR-TR-2005-101*.

S. Osinski and D. Weiss. 2005. A concept-driven algorithm for clustering search results. *Intelligent Systems, IEEE*, 20(3):48–54.

Y. Park, R.J. Byrd, and B.K. Boguraev. 2002. Automatic glossary extraction: beyond terminology identification. In *Proceedings of COLING 2002*, pages 1–7. Association for Computational Linguistics.

Christian Plaunt and Barbara A Norgard. 1998. An Association Based Method for Automatic Indexing with a Controlled Vocabulary. *Journal of the American Society for Information Science and Technology*, 49(10):888–902.

S. Rose, D. Engel, N. Cramer, and W. Cowley. 2010. Automatic keyword extraction from individual documents. In Michael W. Berry and Jacob Kogan, editors, *Text Mining: Applications and Theory*, pages 3–20. John Wiley & Sons.

P.D. Turney. 2000. Learning algorithms for keyphrase extraction. *Information Retrieval*, 2(4):303–336.

A. van den Bosch, G.J. Busser, W. Daelemans, and S Canisius. 2007. An efficient memory-based morphosyntactic tagger and parser for Dutch. In F. van Eynde, P. Dirix, I. Schuurman, and V. Vandeghinste, editors, *Selected Papers of the 17th Computational Linguistics in the Netherlands Meeting*, pages 99–114, Leuven, Belgium.

S. Xu, S. Yang, and F.C.M. Lau. 2010. Keyword extraction and headline generation using novel word features. *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-10)*.

W. Yih, J. Goodman, and V.R. Carvalho. 2006. Finding advertising keywords on web pages. In *Proceedings of the 15th international conference on World Wide Web*, pages 213–222. ACM.

K. Zhang, H. Xu, J. Tang, and J. Li. 2006. Keyword extraction using support vector machine. *Advances in Web-Age Information Management*, pages 85–96.

C. Zhang, H. Wang, Y. Liu, D. Wu, Y. Liao, and B. Wang. 2008. Automatic keyword extraction from documents using conditional random fields. *Journal of Computational Information Systems*, 4(3):1169–1180.