periodica polytechnica

Electrical Engineering
54/3-4 (2010) 93–101
doi: 10.3311/pp.ee.2010-3-4.02
web: http://www.pp.bme.hu/ee

RESEARCH ARTICLE

Topic comparison of remote documents using small communication traffic

Kristóf Csorba / *István* Vajk

Received 2011-09-13

Abstract

This paper presents a new method for semantic search solutions designed for mobile device environments. The proposed system aims at helping users by searching for documents which have similar topics to the ones stored on the users own device. The search is performed in background continuously and the user is notified if documents worth for downloading were found. The methods proposed in this paper aim at solving this task while maintaining low communication traffic to make them applicable in the mobile device environment.

Keywords

document classification · keyword selection · mobile device · compact representation · semantic search

Acknowledgement

This work has been funded by the Hungarian Academy of Sciences for control research and the Hungarian National Research Fund (grant number T68370).

Kristóf Csorba
István Vajk
Department of Automation and Applied Informatics, BME, H-1117, Magyar Tudósok körútja 2., Hungary

1 Introduction

Nowadays it is a rather common application that someone stores documents, like e-books for instance, on a PDA or on a mobile phone. As the user is assumed to be interested in the topic of the own documents [4] [2], a system searching for similar documents to the locally stored ones would be reasonable. This paper proposes such a system: documents which might be interesting for the user are searched for using a background process which keeps on comparing the remote public documents to the local ones. The user is only notified if a document similar to the local ones is found. There is no need to enter search phrases as in the traditional searching cases.

Communication traffic is of key importance between mobile devices: on one hand the communication might not be available free of charge and on the other hand continuous communication depletes the batteries of the devices in a few hours.

The first question is how to represent a document to allow topic comparison while keeping the size of the representation as small as possible. Document comparison techniques usually use the bag-of-words approach which represents the documents initially by vectors in which a unique dimension is assigned to every possible word. This often leads to dimension numbers higher than 10,000. As these vectors are very sparse many dimensionality reduction techniques have been proposed to reduce the feature space by trying to capture semantic relationships between the individual words. Many common approaches use feature extraction techniques which derive new features based on transformations of the original feature space. Common examples are singular value decomposition [11], orthogonal locality preserving indexing [3], double clustering [19] or latent semantic indexing and Rocchio relevance feedback [7]. Using such a technique would not only require to handle huge (probably floating point) feature space transformation matrices and document vectors but also to store a complete list of words with dimension index assignments for the generation of the document vectors.

Feature selection techniques [1, 17, 18] avoid the feature space transformation as they select some of the original features and discard the remaining ones. Examples are information gain or mutual information-based feature selections [16], opti-

mal orthogonal centroid feature selection [20], keyword selection based on document vector centroids [10], and lexical chain based keyword selection [8]. The drawback of these techniques in the current application is that these techniques may assign negative weights: the proposed topic representation does not allow weighting which prevents the application of words with negative weight. For example if there are two topics, one about *animals* and one about *nuclear physics*, the keyword *animal* could get strong negative weight in the topic *nuclear physics*.

The first part of the contribution is a new feature selection method which allows creating a very compact and still easily comparable topic representation of the documents. The size of the compact topic representation is 1 bit/feature (plus some additional information with constant overhead) which leads to 20-30 bytes/document. A disadvantage of the 1 bit/feature representation is the absence of feature weights: features which would have to get negative weight cannot be used. This motivates the proposed feature selection method because most common feature selection techniques are not applicable.

Using topic specific keywords has an important drawback: there are many document pairs with related topic but no common keywords. As the proposed similarity measure is based on common keywords, two documents without common keywords have zero similarity. But a document about *hawks* and one about *dolphins* should have higher similarity than the document about *hawks* and one about *space research* even if there are no common keywords. The second part of the contribution of this paper is a new document extension technique which aims at improving the solution by adding generalizing keywords to the documents. For example the keyword *animal* could be added to the documents about *hawks* and *dolphins* which would lead to non-zero similarity. This solution is similar to query expansion techniques [15]. To recognize the generalization relationship from *dolphin* to *animal* two new techniques are proposed: one is using WordNet [9] and the other one is an unsupervised learning technique. The result of the learning of generalizing words is similar to an ontology or concept hierarchy [5].

The remaining part of the paper is organized as follows: Section 2 presents the feature selection, document topic representation, and topic similarity measurement. Section 3 presents the document extension, Section 4 presents several experimental results and conclusions are drawn in Section 5.

2 Identification, representation and comparison of document topics

This section describes how the document representation of a previously unseen document is created and how to find documents with similar topic. The key idea of the document representation is the following: first the topic of the document is identified which will be indicated by a topic identifier number. This topic identifier could be enough for the search for similar documents but topic assignments are too rough for the current application: keyword based similarity allows finer comparison than just the topic identifier of the documents. Using a keyword list containing typical keywords from the topic of the document, a *binary mask* (binary vector) indicating the presence or absence of these keywords in the document is created. Similarity of two documents is measured using the number of common keywords which can be calculated using these binary vectors. As all words belong to unique dimensions in a *global word space*, the similarity measure is the inner product of the binary document representation vectors.

2.1 Notations

The following notations are used in the discussions:

- T: a document topic, handled as a set of documents. If multiple topics have to be distinguished, other capital letters are used. A document topic might be *sports* for example.
- $\mathcal{T}(d)$: function, returning the real topic of document d, its result is a document set, $d \in \mathcal{T}(d)$.
- d: arbitrary document, a set of words.
- *Base documents*: documents stored locally on the users own mobile device.
- $\mathbf{d}$: original binary document vector of document d. It indicates only the presence/absence of the words. Unless otherwise noted, all document vectors are represented in the global word space where every possible word is assigned a unique dimension.
- A *keyword* is a word which is present in at least one keyword list. This means that it is a word used in the document representations. A keyword for the topic *sports* could be *championship* for example.
- $K_T = \{w_1, w_2, ..., w_n\}$ is the keyword list (set of keywords w_i) for the topic T.
- *Selector*: A selector S_T is a special binary classifier aiming at selecting documents of a given target topic T. The expression *classifier* is not used because selectors are designed to select documents of a given target topic and not to identify the topic of a document. If a document is not selected by a selector, the topic of the document remains unknown. (This approach is also called one-class classifier.) If $S_T(d)$ is true, the document d is selected by the selector of topic T.
- Two topics G and H are related exactly if they are subtopics of the same topic.

For the sake of simplicity a two-level hierarchy of the topics is assumed with *upper level topics* and their subtopics, the *lower level topics*. It should be noted that the definition of related topics can be generalized using a transitive subtopic relation but that makes a depth limit for relatedness necessary otherwise all topics could be related using the root topic of the hierarchy.

Documents of related topics are expected to be different enough to have few or no common keywords but still have some kind of similarity in the topic. For example if there is a topic *sports* and it has two subtopics, *hockey* and *football*, then hockey and football are related topics. The new document extension technique is meant to support the discovery of such loosely related documents.

For performance measurements the common measures *precision*, *recall*, and *F-measure* are used. If documents of a given topic have to be selected, c is the number of correctly selected documents, f is the number of false selections, and t is the number of documents in the target topic, than precision is defined as $p = c/(c + f)$, recall is $r = c/t$, and F-measure is $f = 2pr/(p + r)$. Precision is a statistic to retrieve the probability of the target topic given a document was selected, that is, $Pr\big(\mathcal{T}(d) = T|S_T(d)\big)$. Precision is used in the probabilistic meaning as well but exact values are always estimations of the probability using the statistic.

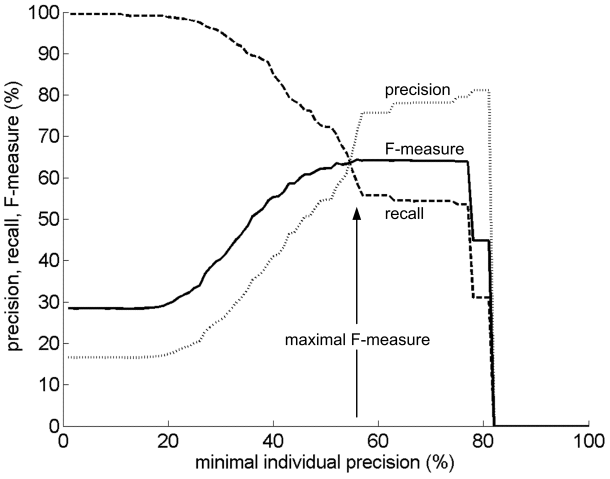

Fig. 1. Decrementing the minimal precision limit (starting from 100%) decreases precision but the increasing number of keywords increases recall. Zero values near 100% indicate that no keywords were found and so no documents could be selected.

2.2 Creating topic specific keyword lists

The proposed compact document representation requires the selection of topic specific keywords for every possible topic. These are words which are very rare in documents of other topics so their presence can be used to identify the topic of the document. The *Precision-based Keyword Selection (PKS)* is a greedy algorithm that aims at creating a keyword list for a given (target) topic using a labeled training set. The algorithm is greedy because it selects always the currently best word to be an additional keyword.

In order to find the keywords specific to a given target topic the *individual precision* of words is defined:

Definition 1 (individual precision) *Given a T target topic, $ip(w)$ individual precision of a word w is the estimated value of the probability $Pr\big(\mathcal{T}(d) = T|w \in d\big)$.*

The expected individual precision for a w word is retrieved using a selector which selects the documents for the target topic T by selecting the documents containing the word w. The expected precision (and so $ip(w)$) is estimated with the measured precision of the selection.

Keywords selected by the PKS algorithm have an individual precision higher than a given minimal limit.

Definition 2 (minimal precision limit) *Given a topic T, the $minprec_T$ minimal precision limit is the minimal individual precision a w word must have to be a keyword of topic T. That is, $w \in K_T \leftrightarrow ip(w) \geq minprec_T$.*

The remaining question is the value of the minimal precision limit. The PKS algorithm iterates through all possible values of $minprec_T$ using 1% steps, creates $K_T(minprec_T)$ and simulates a selector which selects documents containing at least one keyword from K_T. Precision, recall and F-measure of this selection can be measured and finally PKS sets $minprec_T$ to maximize the F-measure of the selection. During the simulation documents containing at least one keyword are selected:

$$|d \cap K_T(minprec_T)| \neq \emptyset \tag{1}$$

where K_T is the keyword list which is a function of $minprec_T$ during the execution of PKS.

Starting from $\text{minprec}_T = 100\%$, the optimization process is shown in Fig. 1. Decreasing minprec_T increases the number of keywords. In the beginning, recall is increasing fast as more and more documents are covered by the keywords. After a while keywords with lower individual precision are added which decreases the precision of the selector through more and more covered off-topic documents. The algorithm is searching for the global maximum of F-measure. After the maximum the increasing recall cannot compensate the falling precision anymore and F-measure begins to decrease.

It is important to note that if there is a topic hierarchy, keyword lists can be created for all the topics on every hierarchy level: both upper and lower level topics can have a keyword list.

A very important property of the keyword lists created by PKS regarding precision and communication traffic is summarized in the following proposition:

Proposition 1 (precision of keyword list) *The keyword list created by PKS allows the highest expected precision for document selection among the keyword lists with the same size.*

The advantage of PKS is that it creates a keyword list suitable for formal propositions on a lower boundary of expected precision but still optimizing for F-measure and so keeping both precision and recall under control. As it has no parameters applications do not have to find optimal settings.

The key idea of the proof, which is not presented in details, is the following: the word w causes a misclassification with a

probability $1 - ip(w)$. Given an n number of words the lowest probability of misclassification and thus highest precision can be achieved by using the n words with the highest individual precisions which is exactly what the PKS algorithm is doing.

2.3 Document topic identification

Creating the document representation of a previously unseen document requires the identification of the topic of the document which identifies the keyword list used for the binary topic representation as well.

The document topic identification is a classification problem. The topic similarity measure is defined as the number of common keywords, so the document representations should preserve as many keywords as possible. This leads to the classification estimating the topic of the document with the topic having the most common keywords present in the document:

$$\widehat{\mathcal{T}}(d) = \underset{T}{\operatorname{argmax}}\{|d \cap K_T|\} \tag{2}$$

Using this classification requires that the presence of a keyword is unlikely in documents from other than its own topic. This is the reason for the minimal individual precision limit in the PKS algorithm. Otherwise a word with low individual precision would increase the number of common keywords in Eq. (2) for off-topics too.

2.4 Document representation

Using the classifier defined previously, the topic of a document can be identified. But a topic, like *animals* for instance, is still too general for the search for similar documents. The key idea for the search is finding documents which share keywords with at least one base document. The representation of a documents topic is derived from the document vector by removing words not present in the keyword list of the estimated topic:

$$(\mathbf{d}_r)_w = 1 \leftrightarrow w \in \mathbf{d} \cap K_{\widehat{\mathcal{T}}(d)} \tag{3}$$

Definition 3 (document representation) *The document representation of a document d is the pair $(\widehat{\mathcal{T}}(d), \mathbf{d}_r)$.*

$\mathbf{d}_r$ can be transmitted by using one bit for only the keywords of the topic as all the other values must be zero and using the topic identifier it is easy to map it into the global word space. If a mobile device does not have the keyword list used in a received document representation, it can be downloaded from a central server.

It should be noted that this type of document representation does not employ document vector normalization. Earlier experiences [6] suggest that if a topic specific keyword appears in a document, then the document is likely to belong to that topic and the exact frequency of the keyword in the document does not hold significant additional information. This observation leaded to the representation using one single indicator bit for every keyword.

2.5 Searching for similar documents

Using the document representation described before the similarity of local and remote documents can be calculated without downloading the whole remote document, only its compact topic representation. The user is notified if a remote document has at least th (threshold) common keywords with the base documents stored on the user's own device. To calculate the number of common keywords a $\mathbf{b}$ binary vector in the global word space is maintained which indicates every keyword of all base documents. If B is the set of base documents and $\mathbf{r}_r$ is the topic representation of the remote document,

$$\mathbf{b} = \operatorname{sign}\Big(\sum_{d \in B} \mathbf{d}_r\Big) \tag{4}$$

where *sign* is the signum function and the user is notified if

$$\mathbf{r}_r^T \mathbf{b} \geq th. \tag{5}$$

The exact value of th is supposed to be set by the user choosing from easy-to-understand options as *many documents* or *strict similarity*.

The following proposition states a lower boundary for the expected precision of the search for similar documents using keyword lists created with PKS.

Proposition 2 (precision of document search) *When searching for documents similar to a given d document with topic T, the minimal precision limit $minprec_T$ used for the creation of K_T is a lower bound for the expected precision of the selection assuming that $\hat{\mathcal{T}}(d) = \mathcal{T}(d)$.*

The condition $\hat{\mathcal{T}}(d) = \mathcal{T}(d)$ ensures that the document representation $\mathbf{d}_r$ is based on the keyword list K_T belonging to the real topic of the document. This proposition states that for example if the keyword list for the topic *animals* was created with $minprec_{animals} = 0.9$ then documents found to be similar to a local document about animals will have the same topic with an expected probability of at least 90%.

The key idea of the formal proof, which is not presented in details, is the following: if the topic of a local document d is identified correctly, that is, assigned a suitable keyword list, the keywords representing it are very rare in documents of other topics. As all the keywords have an individual precision at least *minprec* it can be proven that remote documents r having common keywords with d have the same topic with a probability of at least *minprec*: $Pr(r \in T | w \in d, w \in r) \geq minprec$ because $Pr(d \in T | w \in d) \geq minprec$ due to the definition of the PKS algorithm selecting the employed keywords. This proposition gives an upper bound for the rate of misclassifications during the search for documents similar to d. It can be shown that the same idea is applicable to the $\mathbf{b}$ vector representing all base documents as well by considering the search based on $\mathbf{b}$ as multiple searches based on the individual base documents after each other.

It can be shown that the topic estimation using the number of common keywords between the document and the topics keyword list is very close to the maximum likelihood estimation. It is equal if the $minprec$ values of all the keyword lists are the same.

3 Document extension

A system using the method presented in the previous section can search for documents similar to the base documents. Unfortunately if two documents had related topic, like *dolphins* and *hawks*, but they were not sharing any common keywords, their similarity measure would be zero and they would be considered to be completely different just as any other documents with entirely different topics. The *document extension* procedure slightly increases the similarity measure of documents which have related topics. If the system recognizes *animal* to be a *related generalizing concept (RGC)* to both *hawks* and *dolphins*, the keyword *animal* can be added to both document representations rendering the similarity higher than zero. This would allow finding loosely related documents too.

To achieve this, RGCs of all keywords in a given document are added to the document using the *Related General Concept Function (RGCF)*:

Definition 4 (Related General Concepts Function (RGCF)) *$RGCF$ is the function returning the set of related general concepts (keywords) v_i for the keyword w:* $RGCF(w) = \{v_1, v_2, ..., v_n\}$.

It should be noted that the RGC-s have to be keywords as well, otherwise their addition could not be indicated in the document topic representations.

The extended topic representation d^{ext} of a document d is created as

$$d^{ext} = d \cup \bigcup_{w \in d} RGCF(w). \tag{6}$$

Two ways for creating the RGCF are presented in the following: an unsupervised RGCF learning method and a WordNet based approach.

3.1 Unsupervised RGCF learning

If two documents have a similar upper level topic (for example both are about animals), they are assumed to tend to contain lower level topic specific keywords for their own topic, like *hawk* and *dolphin*, but they also often contain more general keywords from keyword lists of upper level topics such as *animal*. This observation suggests that both *hawk* and *dolphin* are related to *animal* but *animal* is the keyword of an upper level topic which means it is specific to something more general than *hawk* and *dolphin*. If a word is keyword of an upper level topic that means that it is very specific to that upper level topic and it is more general than the keywords of the lower level topics.

The RGCF function returns keywords satisfying the following condition:

$$v \in RGCF(w) \quad \leftrightarrow \quad w \in K_G, v \in K_H : G \subseteq H \wedge \frac{D(w) \cap D(v)}{D(w)} \geq mcr \tag{7}$$

where K_G and K_H are the set of keywords for topics G and H respectively, $G \subseteq H$ indicates that G is a subset of H, $D(w)$ is the set of documents containing the word w, and mcr is the minimal co-occurrence rate (like the minimal confidence limit in association rule mining). The first condition ensures the generalization and the second ensures the frequent co-occurrence of the keywords v and w.

Using these conditions the RGCF can be learned by collecting the RGC-s for every keyword in an unsupervised manner.

The most important feature of the document extension using the unsupervised RGCF learning approach is summarized in the following proposition:

Proposition 3 (probability of adding common keywords) *The probability that a $v \in K_A$ keyword is added to two documents d and f during the document extension (which would increase their similarity measure) is higher if d and f belong to related topics.*

This proposition can be proven based on the observation that a new common keyword is added to the documents only if they have topic specific keywords of related topic. Containing such keywords without belonging to related topics has low probability.

If the same keyword is added to two documents, the similarity measure of that two documents will increase. The aim of document extension is to increase the similarity of documents with related topics first of all in the case of document pairs not having common keywords originally.

It should be noted that learning the RGCF with the statistical method requires the availability of a topic hierarchy which allows using upper and lower level topics. The learned RGCF depends on this, so using different topic hierarchies lead to different Related Generalizing Concept Functions. As the generalizations of the keywords are learned in the context represented by the hierarchy, so if the hierarchy is considered to represent some kind of user preference of topic categorization, the document extension will take this preference into account.

3.2 Creating RGCF using WordNet

Creating the RGCF using WordNet is much easier because WordNet already contains hypernym edges which indicate the generalizations (hypernyms) of the words. In this case, $RGCF(w)$ contains all synonyms and hypernyms of w in a transitive manner using a predefined distance limit. For example if the limit is 2, $RGCF(w)$ contains all words in the symset (set of synonyms in WordNet) of w, and all words in the symsets

achievable through ways along hypernym edges with a maximum length (distance) of 2 edges.

As the document topic representations can indicate only keywords, words returned by WordNet which are not keywords of any topics were omitted.

4 Experimental results

This section presents measurements which evaluate the capabilities of the techniques presented in the previous sections. The measurements were performed using the commonly used data sets 20 Newsgroups [13], RCV1 (LYRL2004 split) [14], and Ohsumed [12].

4.1 Classification measurements

In order to have an overview on the complexity of the classification problem, multiple baseline measurements were performed. Two feature selection methods are compared: PKS and a mutual information (abbreviated as MutInf) based keyword selection which selects a given number of words having the highest mutual information with the topic of the documents. The proposed classification method (abbreviated as MostWords) is compared to a naive bayes (NB) classifier. Results are presented in Table 1. As the MutInf feature selection cannot select an optimal number of keywords, the results using the word numbers optimized by PKS are presented and the maximal F-measure (achievable with word numbers between 1 and 500) is shown in brackets. The mean keyword number per topic returned by PKS is the following: 237.88 for RCV1, 45.60 for 20 Newsgroups and 39.70 for Ohsumed.

It is clear that PKS significantly increases the precision and achieves higher F-measure with both the naive bayes classifier and the proposed MostWords method. Using PKS the proposed MostWords method achieves significantly higher precision and slightly lower F-measure than the naive bayes classifier. For the small decrement in F-measure we get a significant advantage (beside the higher precision): using the proposed MostWords classifier there is no need to transfer and store the weight vectors introduced by the naive bayes classifier, only the keyword lists themselves.

4.2 Keyword list creation

The effectiveness of PKS is indicated by the results of the other methods because all of them operate on document representations based on keyword lists created by PKS. Examples on the keyword list sizes and the minimal precision limits are presented in Table 2. Document extension requires all possible RGCs be in the lower level keyword lists as well so the length of upper level keyword lists contains the length of the lower level keyword lists too. The topic identifier is assumed to be 16 bits. Document representations of these sizes are acceptable in most scenarios. Based on the measurements, topic *sci.electronics* has relative few words with high individual precision which increases the keyword list size and decreases $minprec$. Examples on keywords are presented in Table 3.

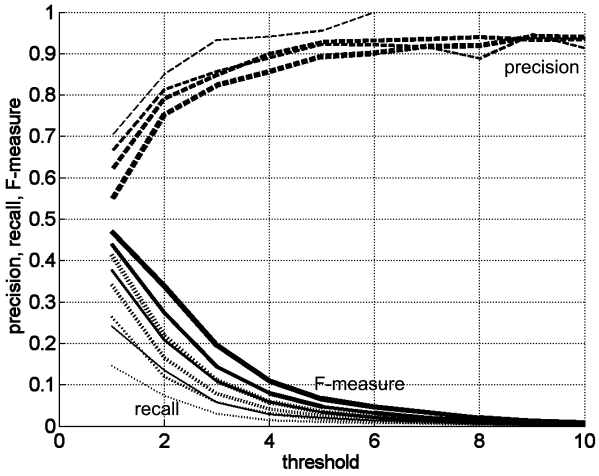

Fig. 2. Evaluation of the search for similar documents in the data set 20 Newsgroups *without* document extension. The precision and recall of the selection is presented as a function of the threshold for base document numbers 1, 5, 10, and 20 (represented by line width in increasing order respectively). Results were evaluated on the upper level topics.

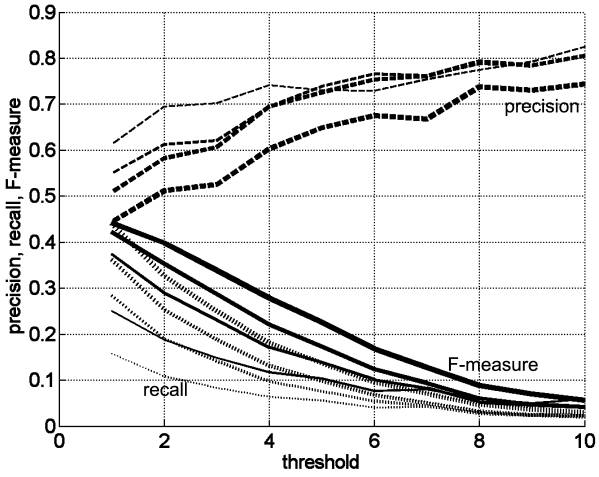

Fig. 3. Evaluation of the search for similar documents in the data set 20 Newsgroups *with* document extension using WordNet based RGCF learning with depth limit 1. Results were evaluated on the upper level topics.

4.3 Learning the Related General Concepts Function

The $RGCF$ learning methods collect the RGC keywords for every keyword of lower level topics. 4 RGCF learning cases are investigated: the unsupervised RGCF learning method and the WordNet based method using distance limits 0, 1 and 2. Table 4 summarizes some example words and their generalizations according to the various cases. It is clear that all the methods capture correct generalizations in some sense but the difference in the operation is clearly visible: the unsupervised method observes co-occurring words and does not take any meanings into account. This leads to topic dependent generalizations which really belong to the topic of the word (like *game* for *players*).

Tab. 1. Classification results using various data sets in terms of precision, recall and F-measure. 20NG stands for 20 Newsgroups and OHS stands for Ohsumed data set.

measurement	dataset	precision	recall	F-measure
MutInf+NB	RCV1	0.3541	0.4919	0.4118 (max 0.42)
PKS+NB	RCV1	0.4600	0.5161	0.4864
PKS+MostWords	RCV1	0.6355	0.4100	0.4746
MutInf+NB	20NG	0.4296	0.5656	0.4883 (max 0.5)
PKS+NB	20NG	0.4781	0.5395	0.5070
PKS+MostWords	20NG	0.6111	0.4543	0.4970
MutInf+NB	OHS	0.3477	0.2789	0.3095 (max. 0.35)
PKS+NB	OHS	0.3628	0.5266	0.4296
PKS+MostWords	OHS	0.4342	0.4078	0.4065

Tab. 2. Size and minimal precision limit of the keyword lists for some upper and lower level topics in 20 Newsgroups. The examples contain both the cases with the least (sport.hockey) and the most keywords (sci.electronics) too.

	number of keywords	$minprec$ (%)	size of representation (bit)
upper level			
hardware	32	81	$16 + 32 = 48$ (6 byte)
sport	21	90	$16 + 21 = 37$ (5 byte)
science	107	73	$16 + 107 = 123$ (7 byte)
lower level			
hardware.pc	33	65	$16 + 33 + 32 = 81$ (11 byte)
hardware.mac	34	56	$16 + 34 + 32 = 82$ (11 byte)
sport.baseball	32	60	$16 + 32 + 21 = 69$ (9 byte)
sport.hockey	26	75	$16 + 26 + 21 = 63$ (8 byte)
sci.electronics	116	48	$16 + 116 + 107 = 239$ (30 byte)
sci.space	107	78	$16 + 107 + 107 = 230$ (29 byte)

On the other hand the WordNet based approach captures generalizations based on real meaning and considers the current topic only so far that the generalization has to be a selected keyword as well. This leads sometimes to generalizations belonging to another sense of the word (like *soul* for *players*). The unsupervised approach seems to be more robust against special words (often not known by WordNet). For example WordNet does not know about *NHL* and so it returns no further generalizations. The unsupervised method recognized that *NHL* is a team game.

Regarding the RGCF learning methods in general we believe that document pairs containing the mentioned generalizations have related topic with high probability. This does not necessarily mean topic equivalence but indicates a little more similarity than nothing which would be characterized by zero common keywords.

4.4 Searching for similar documents

The search for similar documents is evaluated together with the document extension in the following way: a small set of documents is selected from an arbitrary lower level topic and they are considered the base documents. Using these documents a searching for similar documents is started on the remaining part of the testing document set. This procedure is intended to simulate the scenario, where a set of base documents is present on the users mobile device and the device is searching for remote documents (among the remaining elements of the test document set in this case). As documents of related topics are considered loosely similar, the resulting set of selected documents is evaluated for precision and recall using their upper level topics only. By using lower level topics for the evaluation, document extension would drastically decrease the precision by making documents from other lower level topics similar. Finding more documents with related topics is the aim of the document extension.

Fig. 2 shows the results of a search for similar documents using the original document representations and Fig. 3 presents the results with extended document representations. The threshold (minimal number of common keywords for selection) is defined by the user. The measurements are performed for various base document numbers (1, 5, 10, 15 and 20 base documents), and the precision and recall of the simulated search is calculated in a function of the threshold.

The results confirm that the proposed methods maintain a high precision prior to a high recall to minimize the probability of false notifications. Increasing the threshold obviously increases the precision and lowers the recall. Intuitive threshold settings such as "many documents" and "strict similarity" could mean threshold values for example 1 and 3 respectively. The increasing number of base documents increases the set of used keywords thus it increases the recall, but it makes more chances for misclassifications which lowers the precision. The significant recall increment due to document extension is confirmed by the results. This is a consequence of increasing the number of

Tab. 3. Example for keywords representing the topic of a concrete document about space shuttles.

earth, access, protection, mass, landing, os, proposed, schedule, km, planned, fly, adams, bursts, evidence, orbital, space, universe, electrical, mars, predict, earth, vehicle, houston, training, scientific, baltimore, gravity, human, receiver, propulsion, thermal, engines, stanford, sky, satellite, nasa, mission, flight, bases, air, age, rocket, planets, launched, safety, solar, flight...

Tab. 4. Comparison of RGCF learning results. Unsupervised stands for the unsupervised method and WN*n* stands for the WordNet based methods where *n* is the distance limit.

original word	unsupervised	WN0	WN1	WN2
graphics	graphics	art graphics	art graphics	art graphics
controller	controller mb	control controller	person someone individual control soul somebody controller	person being someone cause control individual soul device somebody controller
players	game team player	players player	players player	person soul someone individual somebody players player
encryption	encryption key secure chip keys	encryption	encryption	writing encryption
nhl	nhl game team	nhl	nhl	nhl
ball	ball game	ball	ball baseball shot	ball party equipment baseball shot throw player
cup	cup team	cup	cup hole	cup solid hole

loosely related documents having similarity measure above the threshold. The degradation of precision is acceptable for small base document numbers. For more base documents a stronger precision decrement is observable which is caused by the added RGC keywords and their additional chance to cause false selection. This can be compensated by increasing the threshold if many documents are stored on the mobile device.

Fig. 4 presents a comparison of mean performances of the searches using document extension with various RGCF learning methods or no document extension at all. More additional keywords obviously increase recall and decrease precision. The unsupervised RGCF learning allows slightly higher precision than adding the synonyms based on WordNet.

Table 3 showed previously the keywords of a concrete document about space shuttles. Table 5 shows the keywords added to that document during document extension using unsupervised RGCF learning. Documents containing the additional keywords might have a *loose relationship* to the content of the original document. Although the presence of these keywords alone may imply non-zero similarity measure to many other documents, this should still not be considered to be a clear indication of topic similarity. Strong similarity is indicated by multiple common keywords usually present in the original document vectors.

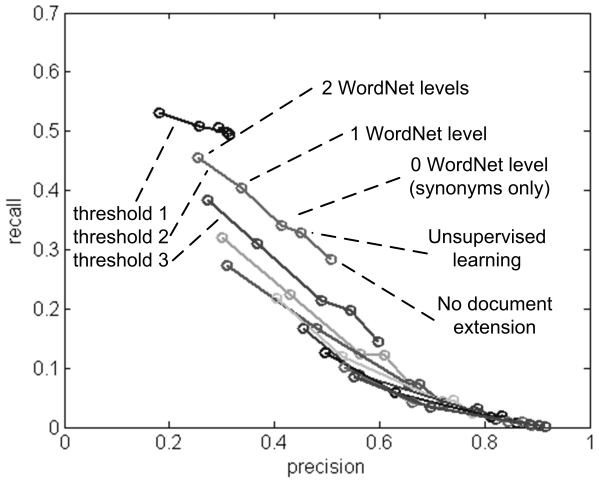

Fig. 4. Performance of similar document search in terms of precision and recall using various RGCF learning methods and threshold values between 1 and 10 on the 20 Newsgroups data set.

Tab. 5. Keywords added to the representation of the document on space shuttles during document extension using unsupervised RGCF learning.

project, sci, phase, science, elements, objects, probe, radar, fuel, toronto, planet, zoo, cloud, solar, kelvin, henry, antenna, probes

Compared to the baseline measurements these results were achieved by using a classifier based on a few baseline documents and not a classifier trained on the whole training document set. Due to the small knowledge of the classifier about the target topic (only the keywords appearing in the base documents) the recall is worse but the precision is often higher due to the precision based selection of the keywords. But the key result beside the high precision is the size of the document representations: on one hand, based on Table 2, the whole classifier requires around 20-30 bytes per base document which is independent of the number of topics (except the influence of topic numbers on keyword list sizes). On the other hand, the comparison of a remote document to the local ones requires only the transmission of the remote compact document topic representation having the size of about 20-30 bytes.

5 Conclusions

This paper presented a new document representation and topic comparison technique for mobile devices. The various devices can search for remote documents having similar topics to the ones stored on them using a background process and notify the user if they find information that might be of interest. The system keeps the communication traffic low by using very compact document representations. False notifications are considered worse than not finding all interesting documents so the system is optimized primarily on high-precision document selection and only secondarily for high recall. This property is ensured by the keyword selection algorithm which selects only very topic specific keywords.

Documents of related topics but few or no common keywords can be found with the help of the document extension which employs two possible techniques for learning the semantic relationship of keywords while preserving the suitability for high-precision topic comparison.

References

1 **Boger Z, Kuflik T, Shoval P, Shapira B**, *Automatic keyword identification by artificial neural networks compared to manual identification by users of filtering systems*, Information Processing & Management **37** (2001), no. 2, 187–198, DOI 10.1016/S0306-4573(00)00030-3.

2 **Buntine W**, *Topic–specific scoring of documents for relevant retrieval*, Proc. ICML 2005 Workshop 4: Learning in Web Search (2005).

3 **Cai D, He X**, *Orthogonal locality preserving indexing*, Proc. 28th annual international ACM SIGIR conference on Research and development in information retrieval, ACM Press, New York, NY, USA, 2005, pp. 3–10, DOI 10.1145/1076034.1076039.

4 **Chirita P A, Firan C S, Nejdl W**, *Summarizing local context to personalize global web search*, Proc. CIKM '06: 15th ACM international conference on Information and knowledge management, posted on 2006, 287–296, DOI 10.1145/1183614.1183658, (to appear in print).

5 **Cimiano P, Hotho A, Staab S**, *Learning concept hierarchies from text corpora using formal concept analysis*, Journal of Artificial Intelligence Research **24** (2005), 305–339.

6 **Csorba K, Vajk I**, *Transformations and Selection Methods in Document Clustering* (Machado J A Tenreiro, Pátkai Béla, Rudas Imre J., eds.), Springer, 2009.

7 **Efron M**, *Query expansion and dimensionality reduction: Notions of optimality in rocchio relevance feedback and latent semantic indexing*, Information Processing & Management **44** (2008), 163–180, DOI 10.1016/j.ipm.2006.12.008.

8 **Ercan G, Cicekli I**, *Using lexical chains for keyword extraction*, Information Processing & Management **43** (2007), 1705–1714, DOI 10.1016/j.ipm.2007.01.015.

9 **Fellbaum C (ed.)**, *WordNet: An Electronic Lexical Database*, The MIT Press, Cambridge, Massachusetts, 1989.

10 **Fortuna B, Mladenic C, Grobelnik M**, *Semi–automatic construction of topic ontology*, Proc. SIKDD 2005 at multiconference IS 2005, 17 Oct 2005 **1** (2005), DOI 10.1007/11908678_8.

11 **Furnas G., Deerwester S, Dumais S T, Landauer T K, Harshman R, Streeter L A, Lochbaum K E**, *Information retrieval using a singular value decomposition model of latent semantic structure*, Proc. 11th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval **11** (1988), 465–480, DOI 10.1145/62437.62487.

12 **Hersh W R, Buckley C, Leone T J, Hickam D. H.**, *Ohsumed: An interactive retrieval evaluation and large test collection for research*, Proc. ACM SIGIR 1994 Conference **1**, 192–201, DOI 10.1007/978-1-4471-2099-5_20.

13 **Lang K**, NewsWeeder*: learning to filter netnews*, Proc. 12th International Conference on Machine Learning (1995), 331–339.

14 **Lewis D. D., Yang Y, Rose T, Li F**, *RCV1: A New Benchmark Collection for Text Categorization Research*, Journal of Machine Learning Research **5** (2004), 361–397.

15 **Lin H.–C., Wang L H, Chen S M**, *Query expansion for document retrieval based on fuzzy rules and user relevance feedback techniques*, Expert Systems with Applications **31** (2006), 397–405, DOI 10.1016/j.eswa.2005.09.078.

16 **Maimon O, Rokach L (eds.)**, *The Data Mining and Knowledge Discovery Handbook*, Springer, 2005.

17 **Schönhofen P, Charaf H.**, *Using Concept Relationships to Improve Document Categorization*, Periodica Polytechnica Elec. Eng. **48** (2004), no. 3-4, 165–182.

18 **Selamat A, Omatu S**, *Web page feature selection and classification using neural networks*, Information Sciences **158** (2004), 69–88, DOI 10.1016/j.ins.2003.03.003.

19 **Slonim N., Tishby N.**, *Document clustering using word clusters via the information bottleneck method*, Proc. 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Clustering **23** (2000), 208–215, DOI 10.1145/345508.345578.

20 **Yan J, Liu N, Zhang B, Yan S, Chen Z, Cheng Q, Fan W, Ma W Y**, *Ocfs: optimal orthogonal centroid feature selection for text categorization*, Proc. 28th annual international ACM SIGIR conference on Research and development in information retrieval (2005), 122–129.