# Journal of Applied Sciences

ISSN 1812-5654

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Applied Sciences 5 (3): 584-587, 2005
ISSN 1812-5654


# Al-Hadith Text Classifier

[1]Mohammed Naji Al-Kabi, [1]Ghassan Kanaan, [2]Riyad Al-Shalabi,
[2]Saja I. Al- Sinjilawi and [1]Ronza S. Al- Mustafa
[1]Department of Computer Information Systems, [2]Department of Computer Science,
Yarmouk University, P.O. Box 566, 21163 Irbid, Jordan

**Abstract:** This study explore the implementation of a text classification method to classify the prophet Mohammed (PBUH) hadiths (sayings) using Sahih Al-Bukhari classification. The sayings explain the Holy Qur'an, which considered by Muslims to be the direct word of Allah. Present method adopts TF/IDF (Term Frequency-Inverse Document Frequency) which is used usually for text search. TF/IDF was used for term weighting, in which document weights for the selected terms are computed, to classify non-vocalized sayings, after their terms (keywords have been transformed to the corresponding canonical form (i.e., roots), to one of eight Books (classes), according to Al-Bukhari classification. A term would have a higher weight if it were a good descriptor for a particular book, i.e., it appears frequently in the book but is infrequent in the entire corpus. The classifier first uses a training set as a learning phase and then uses the test set to evaluate the accuracy of this classifier; the average accuracy for this sample is approximately 83.2%.

**Key words:** Arabic Text Categorization, hadith (prophet sayings) text classifier, Arabic text classification, Arabic text mining, data-mining, classification

## INTRODUCTION

Automated Text Categorization (ATC) is the task of building software tools capable of classifying text (or hypertext) documents under predefined categories or subject codes. ATC has witnessed a booming interest in recent times, due to the availability of ever larger numbers of text documents in digital form and to the ensuing need to organize them for easier use. The dominant approach is nowadays one of building text classifiers automatically by learning the characteristics of the categories from a training set of pre-classified documents (http://mason.gmu.edu/~kersch/JIIS/Special_Issues/TextCategory.html).

Sahih Bukhari is a collection of sayings and deeds of Prophet Muhammad (PBUH), also known as the Sunnah. The reports of the Prophet's sayings and deeds are called hadith. Hadith consists of two main parts, the Sanad and Matn. Bukhari lived a couple of centuries after the Prophet's death and worked extremely hard to collect his hadith. Each report in his collection was checked for compatibility with the Qur'an and the veracity of the chain of reporters had to be painstakingly established. Bukhari's collection is recognized by the overwhelming majority of the Muslim world to be one of the most authentic collections of the Sunnah of the Prophet (PBUH) (http://www.usc.edu/dept/MSA/fundamentals/hadithsunnah/bukhari/sbtintro.html).

Bukhari spent sixteen years compiling it and ended up with 2,602 hadith without repetition (9,082 with repetition). His criteria for acceptance into the collection were amongst the most stringent of all the scholars of hadith (http://www.usc.edu/dept/MSA/fundamentals/hadithsunnah/bukhari/sbtintro.html).

Each hadith is preceded by a chain of the names of those who have transmitted it in each generation, leading all the way back to the companion who reported it from the Prophet. These isnads (Sanad) guarantee the authenticity and verbal accuracy of hadith. For the first few generations, the hadiths are believed to have been transmitted mainly orally rather than in writing[1].

Many algorithms and technique have been applied for many years to text categorization and classification. They include decision tree learning, Bayesian learning, nearest neighbor learning and artificial neural networks, early such works may be found in Hassan *et al.*[2] and Bensaid *et al.*[3]

A good study comparing document categorization algorithms can be found in Yang and Liu[4]. Also, Hassan *et al.*[2] present experimental results on document clustering and classification achieved on the Arabic corpus using statistical methods.

**Corresponding Author:** Mohammed Naji Al-Kabi, Department of Computer Information Systems, Yarmouk University, P.O. Box 566, 21163 Irbid, Jordan E-mail: mohammednaji@yahoo.com

Concerning Arabic, one automatic categorizer has been reported to have been put under operational use to classify Arabic documents; it is referred to as "Sakhr's categorizer" (http://www.Sakhr.com).

## METHODOLOGY

Our approach depends on extracting the main terms from hadith, computing term frequency; TF/IDF (Term Frequency-Inverse Document Frequency) method was used for text searching, term weighting; in which document weights for the selected terms are computed, to classify non-vocalized sayings, after filtering the inserted hadith.

Our corpus contains 8 books, separated in 8 files (i.e. each book in a file).

Present methodology can be summarized as follows:

1. Open hadith file.
2. Filter hadith file:
   a. Remove Sanad,
   b. Remove Stop-Words, The system use an executable code to remove stop words in filtering process[5].
3. Find terms stem.
4. Divide hadith into terms.
5. Compute the frequency of each term in hadith relative to each book (8 books) and construct a term frequency table.

Table 1: Term frequency table

| | Book | | | |
|---|---|---|---|---|
| Term | Book1 | Book2 | … | Book8 |
| Term1 | $freq_{11}$ | $freq_{12}$ | … | $freq_{18}$ |
| Term2 | $freq_{21}$ | $freq_{22}$ | … | $freq_{28}$ |
| … | … | … | … | … |
| Term n | $freq_{n1}$ | $freq_{n2}$ | … | $freq_{n8}$ |

6. Calculate support (threshold) for each term as frequency of term in all documents / # of documents
7. Determine the interesting terms according to a predetermined threshold.
8. Compute the TF/IDF equation as:

$$w = tf * \log_2 (N / df)$$

Where:

tf: is a term's frequency in the document

df: is the frequency of documents in the corpus that contain the term

N: is the number of documents in the corpus.

9. Calculate the cumulative weights for all terms in each book.
10. Rank the cumulative weights in descending order.
11. Display the highest rank's book.

Table 2: Term frequency table with threshold

| | Book | | | | |
|---|---|---|---|---|---|
| Term | Book1 | Book2 | … | Book8 | Threshold |
| Term1 | $freq_{11}$ | $freq_{12}$ | … | $freq_{18}$ | $Thr_1$ |
| Term2 | $freq_{21}$ | $freq_{22}$ | … | $freq_{28}$ | Thr2 |
| … | … | … | … | … | … |
| Term n | $freq_{n1}$ | $freq_{n2}$ | … | $freq_{n8}$ | $Thr_n$ |

Table 3: Cumulative weight table

| | Book | | | |
|---|---|---|---|---|
| Term | Book1 | Book2 | … | Book8 |
| Term1 | $w_{11}$ | $w_{12}$ | … | $w_{18}$ |
| Term2 | $w_{21}$ | $w_{22}$ | … | $w_{28}$ |
| … | … | … | … | … |
| Term n | $w_{n1}$ | $w_{n2}$ | … | $w_{n8}$ |
| Cumulative weights | $\sum_{i=1}^{n} W_{i1}$ | $\sum_{i=1}^{n} W_{i2}$ | … | $\sum_{i=1}^{n} W_{i8}$ |

## ALGORITHM

The algorithm that is used in this paper was implemented using Microsoft Visual Basic programming Language, such language support the Arabic texts, provide a variety of string functions and deal with files in a helpful way.

Figure 1 shows the sequence of this classification process:

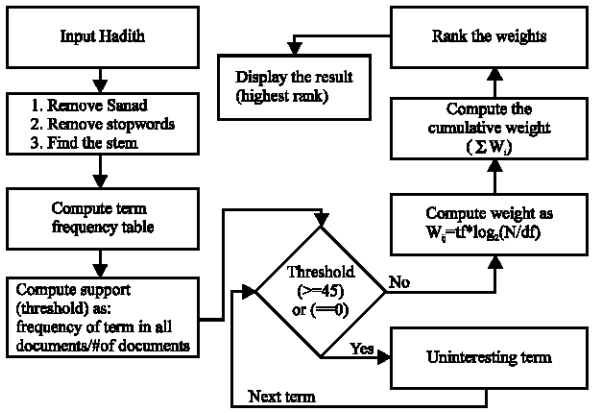


Fig. 1: Classification process

Algorithm: Hadith-Classifying (Input hadith: String, corpus (books) as strings, Output the class with the highest weight of all books).

tf: term frequency

N: number of books.

df: frequency of documents containing term.

(1) Read hadith from a file.
(2) Filtering hadith by
(3) Removing hadith Sanad
(4) Removing stop words.
(5) Initialize the count

(6) For each term in hadith
(7) Count = count + 1
(8) Find term stem
(9) For each book
(10) F= frequency of each stem.
(11) Store F in the term frequency table.
(12) For each entry in the term frequency table
(13) Compute threshold as:
(14) Threshold = frequency of term in all books / # of books.
(15) If threshold >= 45 or threshold = 0
(16) Mark stem as uninteresting term
(17) Else
(18) Compute weight for each *stem* as:
(19) $W_{ij} = tf * \log_2(N / df)$.
(20) End If
(21) Calculate cumulative weights as
(22) For j = 1 to # of Books
(23) $Cum\_weight = \sum_{i=1}^{count} Wij$
(24) Rank or sort results in a descending order
(25) Result = book with the highest rank
(26) Display the result.

## CHALLENGES

Usually natural language projects don't give accurate results. Our system accuracy depends heavily on the accuracy of stop words and stemmer systems, which normally has its own flaws.

We choose hadiths that contain frequent terms; others that do not contain such terms have been skipped (which depends on the semantics of hadith).

In Sahih al-Bukhari we notice that the same hadith may belong to more than one book. Our system can handle such case by displaying two books with the highest ranks.

Our corpus is limited (contains only 8 books), we should enlarge it to contain more books and hadith.

One of the main drawbacks of our system relies in its inability to classify hadiths according to their semantics, so our system cannot classify correctly the following hadith:

عن عدي بن حاتم أن النبي صلى الله عليه وسلم قال :

(اتقوا النار ولو بشق تمره فإن لم تجدوا فبكلمة طيبة)

التصنيف: باب الزكاة

'Adi b. Hatim reported that he heard Allah's Messenger (may peace and blessings be upon him) as saying: "He who among you can protect himself against Fire, he should do so, even if it should be with half a date".

Classification: Almsgiving book

## EVALUATION

In order to test the accuracy of our system, we selected 80 hadiths that resides in 8 books (Fig. 2). Table 4 summarizes the accuracy measures; the average accuracy for this sample is approximately 83.2%.

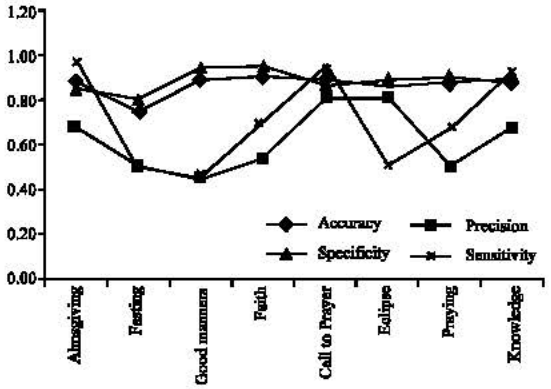


Fig. 2: Accuracy measures


Fig. 3: Main form


Fig. 4: Hadith classification form

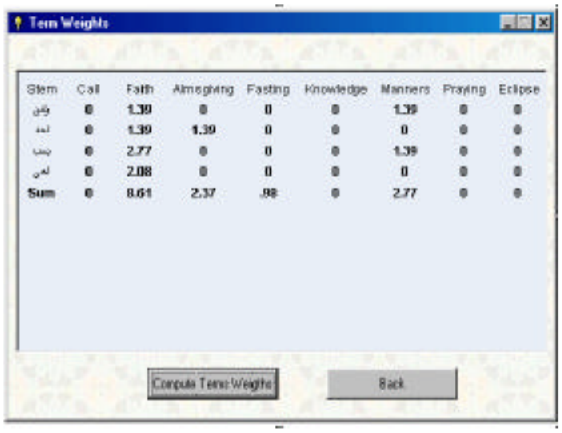


Fig. 5: Term weight computation form

Table 4: Accuracy measures

| الباب | Book | Accuracy | Precision | Specificity | Sensitivity |
|---|---|---|---|---|---|
| العلم | Knowledge | 0.87 | 0.67 | 0.87 | 0.93 |
| الصلاة | Praying | 0.87 | 0.50 | 0.90 | 0.67 |
| الكسوف | Eclipse | 0.86 | 0.80 | 0.89 | 0.50 |
| الآزان | Call to Prayer | 0.87 | 0.80 | 0.86 | 0.96 |
| الايمان | Faith | 0.91 | 0.53 | 0.95 | 0.67 |
| الأدب | Good manners | 0.88 | 0.44 | 0.94 | 0.45 |
| الصوم | Fasting | 0.75 | 0.50 | 0.80 | 0.50 |
| الزكاة | Almsgiving | 0.87 | 0.67 | 0.85 | 0.97 |

As training set we collect about 15 hadiths for each book and 5 hadiths for each test set, normally when training set is large the classifier accuracy goes up.

We can see that each book has its own terms, so the accuracy of the classifier varies from one book to another.

Now, we show the execution of the system based on our algorithm. The inputs to this system are text files of hadiths.

Next, we show the execution of the system based on our algorithm. Figure 3-5 shows the Interface of the system. The interface enable the user to choice the interface language, open hadith file, removing Sanad, stop word, find the stem of terms and then calculate the weight.

## CONCLUSIONS

In this paper we have described the design and successful implementation of a new method suitable for classifying the prophet Mohammed (PBUH) sayings (Hadiths) in Arabic. The method has been implemented using Microsoft Visual Basic 6.0.

Future work will concentrate on enhancing the method so that it can classify nested classification in the same book for each hadith, for example the following hadith can be classified according to all books as in Faith book and according to hadiths in this book, its classifieds as Faith matters book.

عن أبي هريره رضي الله عنه ، عن النبي صلى الله عليه وسلم :

(الأيمان بضع وستون شعبة و الحياء شعبة من الأيمان)

التصنيف: باب الأيمان ، باب أمور الأيمان

It is narrated on the authority of Abu Huraira that the Messenger of Allah (PBUH) said: Faith has over sixty branches and modesty is the branch of faith.

Classification: Faith book, faith matters book.

## REFERENCES

1. Microsoft, Encarta Encyclopedia, 2001. Microsoft Corporation (1993-2000).
2. Hassan, S., H. Ney and J. Zaplo, 2001. Statistical classification methods for Arabic news articles. ACL/EACL 2001 Workshop, Arabic Language Processing: Status and Prospects Toulouse, (France, Friday 6 July 2001), RWTH Aachen, Germany.
3. Bensaid, A., M. EL-Kourdi and T.E. Rachidi, 2004. Automatic Arabic document categorization based on the naive bayes algorithm. Proceedings of Coling 20th Workshop on Computational Approaches to Arabic Script-based Languages, (Geneva, August 23rd-27th, 2004 ).
4. Yang, Y. and X. Liu, 1999. A re-examination of text categorization methods. Proceedings of the Twenty-Second ACM, SIGIR Conference on Research and Development in Information Retrieval, SIGIR'99, (Berkeley, California, USA, August 15-19, 1999), ACM 1-58113-096-1/99/0007, pp: 42-49.
5. Al-Shalabi, R., G. Kanaan, J.M. Jaam, A. Hasnah and E. Hilat, 2004. Stop-word removal algorithm for Arabic language. Proceedings of 1st International Conference on Information and Communication Technologies: from Theory to Applications, ICTTA'04, (Damascus, Syria, April 2004). IEEE-France, pp: 545-550.