Turkish Journal of Electrical Engineering and Computer Sciences

Volume 26 | Number 6 Article 11

1-1-2018

Temporal specificity-based text classification for information retrieval

SHAFIQ UR REHMAN KHAN

MUHAMMD ARSHAD ISLAM

MUHAMMAD ALEEM

MUHAMMAD AZHAR IQBAL

Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2018) 26: 2915 – 2926
© TÜBİTAK
doi:10.3906/elk-1711-136

Temporal specificity-based text classification for information retrieval

Shafiq Ur Rehman KHAN*, **Muhammad Arshad ISLAM**, **Muhammad ALEEM**, **Muhammad Azhar IQBAL**

Department of Computer Science, Faculty of Computing, Capital University of Science and Technology, Islamabad, Pakistan

Received: 15.11.2017 • **Accepted/Published Online:** 26.06.2018 • **Final Version:** 29.11.2018

Abstract: Time is an important aspect in temporal information retrieval (TIR), a subfield of information retrieval (IR). Web search engines like Google or Bing are common examples of IR systems. An important constituent of a search engine is news retrieval, where users present their information needs in the form of temporal queries. Users are usually interested in news documents focusing on a particular time period. Existing search engines rarely fulfill the temporal information requirements as they ignore the temporal information available in the content of news documents, also known as document focus time. Furthermore, information related to multiple time periods in a news document makes the identification of document focus time a challenging task. Therefore, it is necessary to classify news documents based on temporal specificity before it is possible to use the temporal information in the retrieval process. In this study, we formulate the temporal specificity problem as a time-based classification task by classifying news documents into three temporal classes, i.e. high temporal specificity, medium temporal specificity, and low temporal specificity. For such classification, rule-based and temporal specificity score (TSS)-based classification approaches are proposed. In the former approach, news documents are classified using a defined set of rules that are based on temporal features. The later approach classifies news documents based on a TSS score using the temporal features. The results of the proposed techniques are compared with four machine learning classification algorithms: Bayes net, support vector machine, random forest, and decision tree. The results show that the proposed rule-based classifier outperforms the four algorithms by achieving 82% accuracy, whereas TSS classification achieves 77% accuracy.

Key words: Text classification, temporal classification, temporal specificity, temporal information retrieval, specificity score

1. Introduction

Temporal information retrieval (TIR) is a subdomain of information retrieval (IR), which addresses the temporal information requirements of users, specified through temporal queries. The time dimension is scrutinized in numerous IR processes, such as document preprocessing [1-3], query processing [4,5], ranking/retrieval models [6-9], and evaluation metrics. Traditional search engines consider only textual relevance for searching query terms in the indexed documents and producing unsatisfactory outcomes for temporal queries. TIR systems combine both temporal and textual relevance to retrieve both temporally and textually relevant results, matching the user query.

One of the integral components of search engines is the news search system that indexes news, articles, and editorials from different sources worldwide. The time information is very indispensable for news documents

*Correspondence: shaffiqmasud@gmail.com

that could appear in various forms (such as creation time [10], publication time [11], and update time), found in the metadata of the corresponding news webpage. Another important time notation associated with news documents is content time [12], available in the document text. The content time is essential when estimating the focus time of a document. In addition, document focus time relates the document content (describing a specific event) to a particular time period [13]. If a document discusses a single event in time then that document has a single focus time. Moreover, information related to multiple time periods in a news document makes the identification of document focus time a challenging task.

To address this challenge, we introduce a novel concept, temporal specificity, which is defined as a measure that determines how temporally specific the content of the document is. The more document content focuses on a single time period, the more temporally specific it is. The content of a temporally specific document relates to a few distinct points in time. In this research, the news documents are categorized into three classes based on temporal specificity, namely high temporal specificity (HTS), medium temporal specificity (MTS), and low temporal specificity (LTS), as shown in Figure 1. To the best of our knowledge, this is the first study to classify temporal documents with respect to their temporal specificity. Two methods are proposed for this classification. The first method is rule-based, where a set of rules is defined to classify news documents. These rules are based on temporal features that have been extracted from a news corpus. In the second method, a temporal specificity score (TSS) of documents is calculated and used for classification into the aforementioned classes.

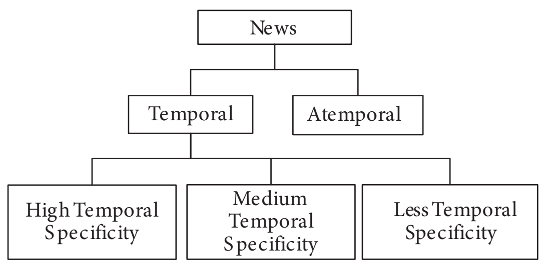

Figure 1. News categorization into temporal, atemporal, HTS, MTS, and LTS.

The rest of the paper is organized as follows. Section 2 presents the motivation for this study. Work related to our research is discussed in Section 3. Section 4 describes the used methodology. Experimental setup and discussion of results are presented in Section 5 and Section 6, respectively. Finally, Section 7 concludes this paper.

2. Motivation

Search engine users face a problem of temporal sparsity in retrieved temporal documents. News documents discussing multiple events do not satisfy the temporal requirements posed by a user query, and commercial search engines do not consider temporal specificity while retrieving documents. We argue that temporal specificity plays an important role in fulfilling user requirements, as described in the following example. Figure 2 presents a timeline of earthquake-related events in Pakistan from 2005 to 2016. These events were covered by leading newspapers in the country. The milestones represent the information related to place and time (year) of the earthquake. In the LTS class, three earthquake events are reported, having a maximum time span of 5 years, shown by the red markers. The blue markers represent MTS news, where three different earthquake events are discussed within a time span of 2 years. Finally, the green markers represent earthquake events that occurred in a single year (2015), and therefore these documents are classified as HTS. These three classes are defined as follows:

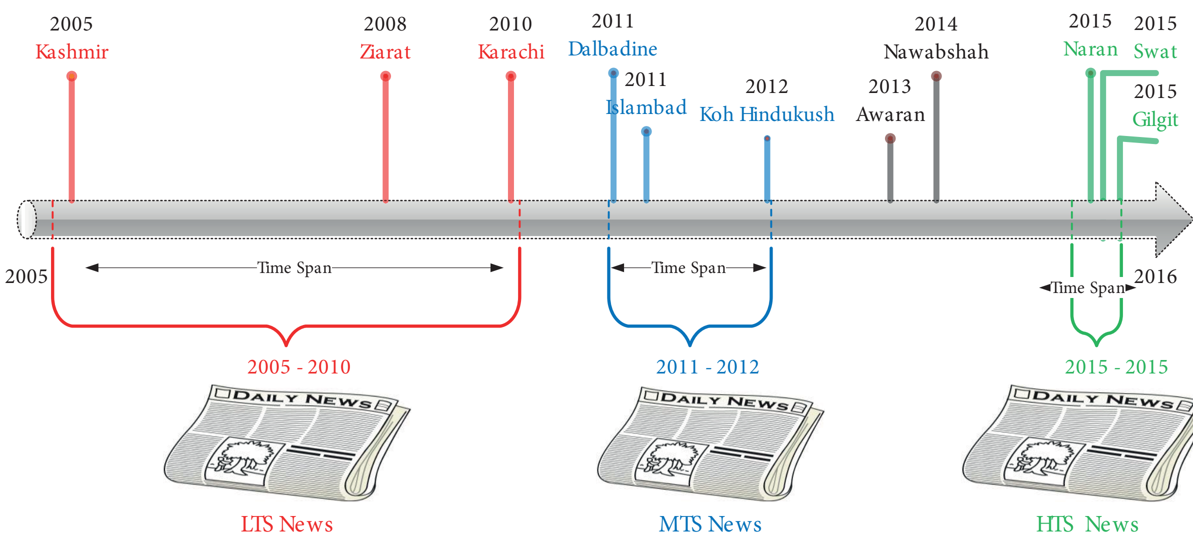

Figure 2. An example of HTS (green), MTS (blue), and LTS (red) news categories on timeline from 2005 to 2015.

HTS: The HTS class refers to a group of articles where one or a few events are discussed within a short time span, focusing more on a single event (single year).

MTS: The MTS class refers to a group of articles where multiple events are discussed within a relatively short time span (3 years).

LTS: The LTS class refers to a group of articles where multiple events are discussed over a long time span (>3 years).

The timespan prescribed for each class has been derived empirically from the dataset. The average document time span observed in the dataset is 2.68 years, as presented in Section 6; therefore, thresholds for LTS, MTS, and HTS are set to >3 years (higher than average time span), 3 years (approximately equal to average time span), and 1 year (less than average time span), respectively.

3. Related work

This research has its roots in TIR and news classification. This section presents the related work in both domains.

3.1. Temporal information retrieval

Alonso et al. [14] highlighted the significance of the time aspect for adequate searching in IR systems. They argued that document creation time is important metadata information: it plays a crucial role in retrieving results for temporal queries, and commercial news search engines tend to rank news based on document creation time. A problem arises when the creation time is not available—what is known as a non-time-stamped document. To address this problem, several methods have been proposed to estimate the creation time of the document, and the process has been called document dating [2,15]. The approach proposed by Alonso [15] classifies documents as content-based and non-content-based. In the former case, the content of the document is used for document dating, which requires an independent time stamp in order to create a temporal language model. In contrast, non-content-based document dating uses external information in the process; the major shortcoming of this method is the unavailability and inaccuracy of external sources. Kanhabua et al. [8] extended the language model with temporal entropy notation, Google Zeitgeist, and semantic preprocessing. Filannino et al. [16] extracted temporal expressions from the text of the document and constructed a timeline associated with the entity (for example, using a Wikipedia page for a person of interest), predicting its upper and lower time boundaries. Niculae et al. [4] used a statistical model to predict the creation date using documents in three languages, English, Portuguese, and Romanian. Implicit temporal queries were investigated in [5,17], which proposed methods for determining the time of implicit temporal queries.

The work of Jatowt et al. [13] has the closest relevance to this paper: they estimated the document focus time through word-time pair association. Words are extracted from articles written at a different point in history and associated with the given time period. If a given document has many words associated with a certain time period t, then the document has a strong association with time period t. Another paper by Spitz et al. [9] proposed a graph-based ranking model, which determines a set of words relevant to a certain time range. They suggested that words and temporal expressions have a strong association if they occur more often at sentence level.

3.2. News classification

The temporal aspect of text classification was investigated by Stanjer and Zampieri [18]. They classified text documents based on the change in writing style and classified Portuguese historical texts into different centuries. Four classification features were used for classification, including average sentence length (ASL), average word length (AWL), lexical density (LD), and lexical richness (LR). The change in the values of these features highlights the creation time of the historical text. Their results revealed that texts written in the 17th and 18th centuries have different AWL, LD, and LR as compared to texts written in the 19th and 20th centuries. Fukumoto and Suzuki [19] proposed a temporal-based feature selection method for document classification. They identified two types of features, named temporally independent terms and temporally dependent terms, in the corpus. For experimental purposes, the documents used as training and testing data have different creation times. The authors applied boosting-based transfer learning to learn an accurate model for timeline adaptation. Comparing their results with a bias support vector machine (SVM) method, they showed improvement in the macro average F-score from 0.671 to 0.688. Luo and Heywood [20] analyzed the temporal sequence of the word and proposed a new method for text representation. Such a new approach was tested for document categorization using the K-nearest neighbor classifier. A micro average F-score of 0.855 was obtained for categorization.

4. Methodology

The process for temporal classification of documents requires the following steps: data cleansing and temporal tagging, temporal profiling, feature extraction, and finally classification, as illustrated in Figure 3.

4.1. Data cleansing and temporal tagging

In this study, we used the Reuters-21578 (https://archive.ics.uci.edu/ml/machine-learning-databases/reuters21578-mld/) dataset in SGML file format. A single file contains multiple news documents tagged with corresponding metadata. The metadata consist of important information such as creation time, title, geographical location, author, and organization. The first step is parsing the news documents using the SGML parser in order to extract the text from the file. The <Date>tag is used as the reference date for temporal tagging. The next step

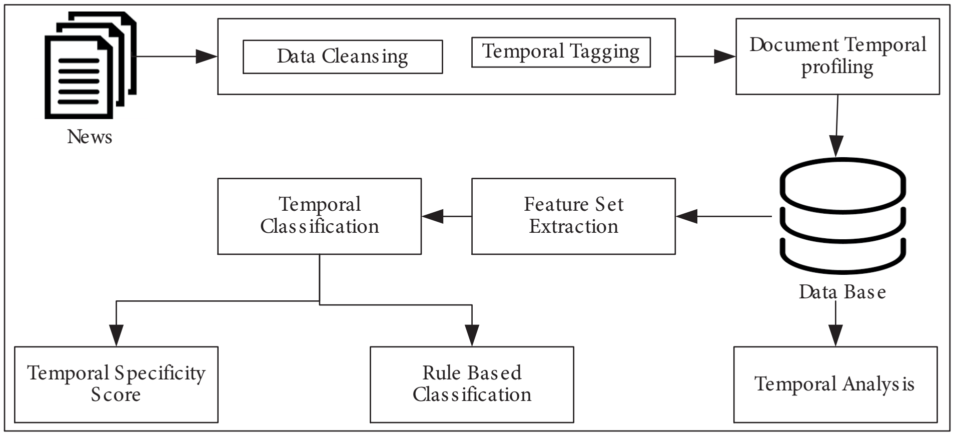

Figure 3. The proposed methodology for temporal classification.

is to remove the unwanted tags from the metadata, i.e. <Title>, <Organization>, and <Location>. Finally, the SGML files are split into multiple text files, each containing a single news item.

After data cleansing, temporal tagging is the process of identifying, extracting, and normalizing temporal expressions in documents [21,22]. We use the HeidelTime [21] temporal tagging library, developed at the University of Heidelberg, Germany, for temporal tagging.

4.2. Document temporal profile

In this step, temporal profiles of news documents are constructed and stored in a database. The temporal profile of an article is represented by the tuple in Eq. (1):

$$TP_d = \{id, te, nte, ny, nm, nd, ct, \varphi\}, \tag{1}$$

where TP_d is a temporal profile of a document d that contains document identifier id, temporal expressions te, normalized temporal expressions nte, normalized years ny, normalized months nm, normalized days nd, and creation time ct. φ represents the set of distinct year frequencies in the document. φ is defined as:

$$\varphi = \{FY_1, FY_2, FY_3, FY_4, \ldots\ldots, FY_n\}, \tag{2}$$

where each FY_i represents the count for the distinct year.

4.3. Temporal feature selection

A total of 43 features for each document in the dataset are extracted, such as mean temporal expression count, mean span of temporal expression, and creation time. The following four features with high information gain have been shortlisted for the analysis presented in this paper.

Distinct year count (Dy) : A document may contain multiple temporal expressions representing multiple distinct years. Dy is the number of total distinct years in the document. The value of Dy is the cardinality of the set φ, i.e. $Dy = |\varphi|$; hence, for each temporal document, $Dy > 0$.

Temporal expression count (FTe): FTe is the count of temporal expressions found in the document after performing temporal tagging. FTe can be represented as:

$$FTe = \sum_{i=1}^{n} FY_i. \tag{3}$$

Maximum likelihood year (ML): Considering Eq. (2), the distinct year having maximum frequency is considered the maximum likelihood year for document d. The likelihood of a distinct year Y_i can be computed as:

$$P(Y_i) = \frac{FY_i}{FTe}, \tag{4}$$

where FY_i is the frequency of the distinct year i, while FTe is the total temporal expression count in the document. This feature is particularly important when the document has multiple distinct years but focuses on a single year. The maximum likelihood year can be calculated as:

$$ML = \max(P(Y_i)) \tag{5}$$

For example, a document d has three distinct years, 1987, 1989, and 1990, and total FTe is equal to 7. The FY values for 1987, 1989, and 1990 are 5, 1, and 1, respectively. Using Eq. (4), each distinct year $P(Y_i)$ is calculated as follows: $P_{1987} = \frac{5}{7} = 0.714$; $P_{1987} = \frac{1}{7} = 0.142$; $P_{1987} = \frac{1}{7} = 0.142$. Hence, ML will be $P_{1987} = \frac{5}{7} = 0.714$, which means that year 1987 is discussed most in the document.

Time span $(Tspan)$: $Tspan$ is the difference between the upper and lower bounds of the interval of temporal expressions (i.e. the upper bound represents the latest year, while the lower bound represents the earliest year mentioned in the document). The time span $Tspan$ for document d is calculated as:

$$Tspan_d = \max T_d - \min T_d \tag{6}$$

where $MaxT_d$ and $MinT_d$ are the respective maximum and minimum time boundaries (in years) of document d. Table 1 illustrates an example of calculating the values for these features.

Table 1. Temporal features calculation example.

Doc ID	Year 1	Fy1	Year 2	Fy2	Year 3	Fy3	Dy	FTe	ML	Tspan
1	1986	2	1988	1	1990	8	3	11	0.72	5
2	1987	1	1990	3	-	-	2	4	0.75	4
3	1997	1	1998	1	2000	1	3	1	0.33	4
4	2001	2	2004	2	2005	4	3	8	0.50	3
5	1966	4	1998	1	1999	1	3	6	0.66	4
6	1985	1	1986	5	-	-	2	6	0.83	2
7	1984	1	-	-	-	-	1	1	1.00	1

5. Proposed classification approaches

For the classification of temporal documents into three classes, we propose two different methods: rule-based classification and TSS-based classification. These two methods are presented in the following discussion.

5.1. Rule-based classification

As discussed earlier, each document in the database has temporal profile TP as shown in Eq. (1). Each document $d \in D$ (D is the document collection) is labeled with class c such that $d \leftarrow c$, where $c \in C$ (C is the set of class labels $C = \{HTS, MTS, LTS\}$). For each class c, a set of rules defined by a condition set are extracted from the annotated data. We extract a set of rules using an association classification method known as classification based on the predicative association rule [24]. The set of rules for each class is selected using foil information gain ($FOIL_{Gain}$) as presented in Eq. (8):

$$FOIL_{Gain}\left(R_0, R_1\right) = P \bullet \left(\log_2 \frac{pos1}{pos1 + neg1}\right) - \left(\log_2 \frac{pos0}{pos0 + negO}\right). \tag{7}$$

$R_0 R_1$ are the rules before and after adding literals, respectively. $pos_0 pos_1 neg_0$, and neg_1 show positive and negative tuples covered by the R_0 and R_1 rules. P represents positive tuples covered by both R_0 and R_1. The rules for each class using the condition set are shown in Table 2.

Table 2. Temporal classification rules with high *Foil_Gain* values are used for rule-based classifier.

R.ID	Rule	Class
1	$Dy \leq 3\, AND\ FTe = 1$	MTS
2	$Dy > 3\, AND\, ML\ \leq 0.67$	LTS
3	$Dy > 3\, AND\, ML > 0.67$	LTS
4	$dy \leq 3\, AND\, FTe > 1\, Tspan \geq 2\ AND\, ML$>0.5	HTS
5	$Dy \leq 3\, AND\, FTe > 1\, AND\ Tspan > 2\ AND\, ML$<0.5	HTS
6	$Dy \leq 3\, AND\, FTe > 1\, AND\ Tspan < 2\, AND\, ML$<0.5	LTS
7	$Dy \leq 3\, AND\, FTe > 1\, AND\ Tspan > 2\, AND\, ML > 0.6$	HTS
8	$Dy \leq 3\, AND\, FTe > 1\, AND\ Tspan > 2\, AND\, ML \leq 0.6$	LTS

5.2. TSS classification

We consider three features that are extracted from the dataset, i.e. DY, FTe, and ML. The TSS score is calculated for each document, where high TSS indicates that the document exhibits HTS whereas a lower TSS score indicates that the document exhibits LTS. We have chosen the threshold value empirically to obtain a suitable classification into three temporal classes for our dataset. The TSS function is presented in Eq. (**??**):

$$TSS = \frac{1}{Dy} \times (ML.FTe) \tag{8}$$

If the frequency of the distinct year (ML) in a document is high, this corresponds to a high TSS and is thus classified as HTS. On the other hand, $\frac{1}{Dy}$ reduces the TSS if the document has multiple distinct years with approximately equal frequencies.

The performance of the proposed methods is compared to four machine learning classification algorithms: Bayes net, SVM, random forest, and decision tree, presented by Eqs. (2)–(6) in Table 3. The results are evaluated using accuracy $A = \frac{K}{N} * 100$, precision $P_c = \frac{K_c}{N_c}$, recall $R_c = \frac{K_c}{R_c}$, and F-score $F_c = 2 \bullet \frac{P_c \bullet R_c}{P_c + R_c}$, where K is the number of correctly classified instances and N is the total number of instances in the dataset D. K_c,

N_c, and R_c represent correctly classified instances, number of instances labeled by the classifier, and number of instances in D, respectively, for class c.

Table 3. Bayes net, SVM, random forest, and decision tree formulae.

Algorithm	Scoring function	Description
Bayes net	$P(U) = \prod_{u \in U} p(u\|pa(u))$ (9)	• Represents probability distribution • U = set of variables • $p(a(u)$ = set of parent in Bayesian network B_s.
Support vector machine (SVM)	$\sum_{i=1}^{m} a_i y^i K\left(x^i, x\right) + b$ (10)	• m = data points • $x^i y^i = i^{th}$ training set point • a_i = coefficient of i^{th} training point, x = input vector • K = kernel function • b = scalar value
Random forest	$Rf = \{DT_1, DT_2, DT_3, \ldots.., DT_n\}$ (11)	• Random forest Rf is a set of decision tree
Decision tree	$Entropy(D) = \sum -p_i * \log_2 P_i$ $Gain(D, A) = Entropy(D)$ $- \sum \left(\frac{\|D_y\|}{\|D\|}\right) * Entropy(D_y)$ $SplitInfo(D, A) = - \sum \left(\frac{\|D_i\|}{\|D\|}\right) * \log_2 \left(\frac{\|S_i\|}{\|S\|}\right)$ $GainRatio(D, A) = \frac{Gain(D,A)}{Splitinfo(D,A)}$ (12)	• D = dataset • p_i = instance belong to class i in D • A = feature • D_y = subset of D. • D_i = resultant subset after splitting D using A

6. Experimental setup

Document collection consisting of 10,000 news articles from Reuters-21578 is used for the temporal analysis. The temporal analysis reveals interesting facts about the temporal information in the news documents as shown in Figures 4 and 5. The selected news articles were published in 1987. After temporal tagging and removal of incorrect annotation, a total of 6703 documents showed temporal expressions and were thus classified as temporal class, whereas the remaining documents are classified as atemporal. From these 6703 documents, 22,983 temporal expressions are extracted, including both explicit and implicit forms. A total of 100 distinct years, ranging from 1887 to 2030, are discussed in selected documents; the frequency distribution of these 100 years is presented in Figure 4. On average, each document contains 3.42 temporal expressions and average time span of 2.68 years. Figure 5 presents the document count with respect to a given number of distinct years.

7. Results and discussion

For the sake of evaluation, we selected 3000 news documents at random from the temporal class and assigned them to multiple annotators, who classified each document into one of the three given classes. For interrater

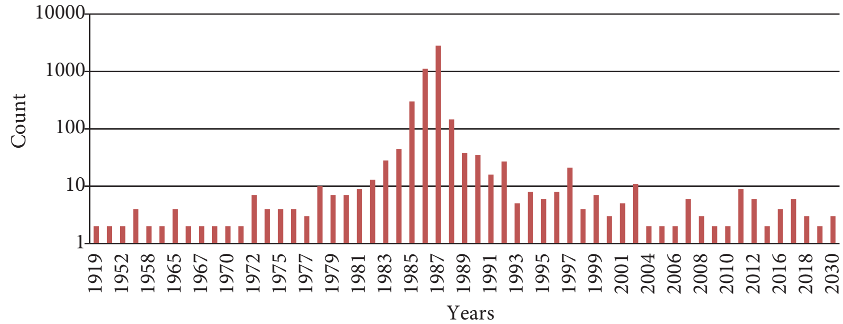

Figure 4. Distinct year count in temporal news documents.

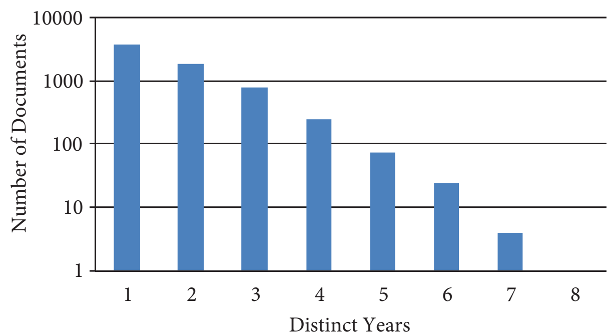

Figure 5. News documents containing number of distinct years.

reliability, Fleiss' kappa [23] is used and calculated as 0.77. The class distribution of selected documents for HTS, MTS, and LTS classes is 1250, 361, and 1389 documents, respectively. We compare the obtained classification results with machine learning classification algorithms: Bayes net (BN), SVM, random forest (RF), and decision tree (DT). All four machine learning algorithms are tested using 10-fold cross-validation. The classification results in term of precision, recall, F-score, and overall accuracy are presented in this paper.

Figure 6 illustrates the overall accuracy of all the classifiers. It can be seen that the rule-based (RB) classifier has the highest accuracy with 82.19% of correctly classified instances, while DT, SVM, RF, and BN have accuracies of 81.72%, 81.49%, 81.19%, and 77.85%, respectively. The TSS classifier correctly classified 77.19% of instances. Figures 7, 8, and 9 depict the results for precision, recall, and F-score, respectively, achieved by each classifier. For the HTS class, it can be shown that the TSS classifier has the highest precision score of 0.92, while the RB classifier achieves 0.87 as shown Figure 7. The performance of BN is the lowest among all classifiers with a precision score of 0.77 for the HTS class. For the MTS class, the RB and TSS classifiers have the lowest precision scores of 0.79 and 0.68, respectively. Finally, for the LTS class, the RB and TSS classifiers reach a precision score of 0.79 and 0.688, respectively.

The BN classifier attains high recall of 0.93 for the HTS class, and for the same class, the RB and TSS classifiers obtain recall scores of 0.83 and 0.74, respectively, as presented in Figure 8. For the MTS class, the RB classifier and the TSS classifier perform well and have 0.83 and 0.94 recall, respectively. BN achieves a high recall value of 0.90, while the TSS classifier attains the lowest recall value of 0.21 for the LTS class.

Figure 9 shows the F-score of the classifiers for each class. For the HTS class, each classifier achieves almost the same result with slight variations. Furthermore, for the MTS class, almost all classifiers have an

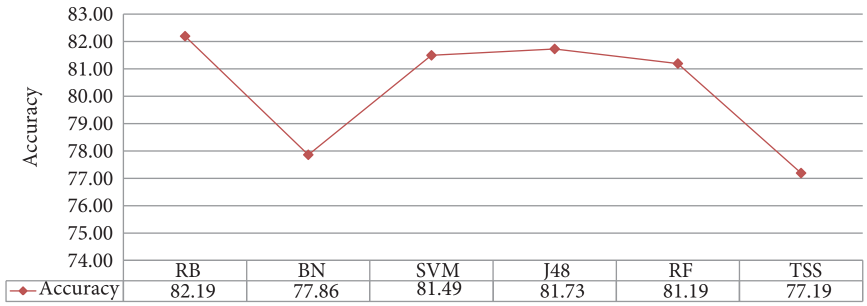

Figure 6. Accuracy score achieved by using six classification techniques.

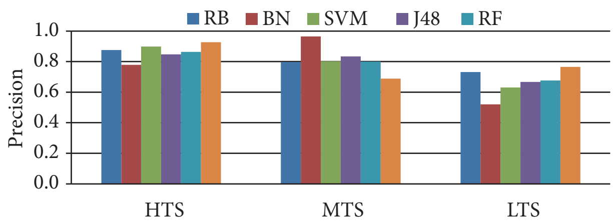

Figure 7. Comparison of classification techniques in terms of precision score.

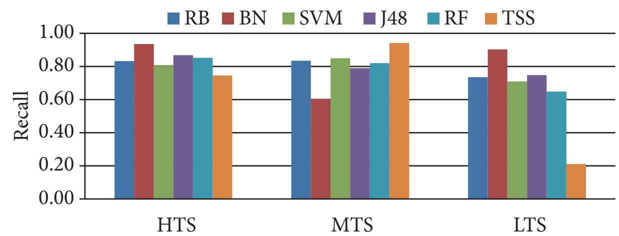

Figure 8. Comparison of classification techniques in terms of recall score.

equal F-score, except for BN, for which the F-score is slightly lower. Finally, for the LTS class, the TSS classifier has the lowest F-score while the RB classifier has the highest F-score of 0.733.

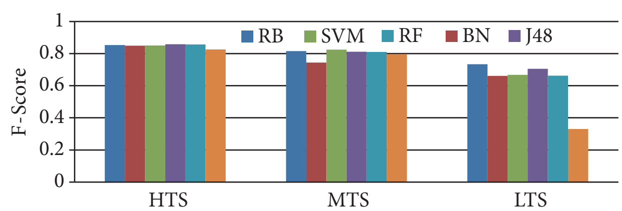

Figure 9. Comparison of classification techniques in terms of F-score.

The experiments reveal that our proposed RB approach outperforms all the other classifiers in terms of overall accuracy. RB and TSS-based classification attains high precision scores for the HTS and LTS classes. In terms of recall, the RB and TSS-based approaches achieve high scores for the MTS class. The RB approach achieves F-score values similar to the other classifiers for the HTS and MTS classes but shows improvement in the LTS class.

All the experiments have been performed on an Intel Core i5-3210 machine with 6 GB RAM running the Windows 8 64-bit operating system using Python 2.7 programming language. The performance of classifiers in terms of the execution time is presented in Table 4. BN has the lowest execution time (0.06 s), followed by TSS (0.13 s), DT (0.28 s), RB (0.64 s), SVM (1.56 s), and RF (2.44 s).

8. Conclusion and future work

In this work, we presented two novel approaches for the classification of news documents with respect to temporal specificity: rule-based classification and TSS-based classification. The news documents are first classified into

Table 4. Execution time taken by the classification algorithms.

Algorithm	Execution time (seconds)
Rule-based	0.64
Bayes net	0.06
SVM	1.56
Decision tree	0.28
Random forest	2.44
TSS	0.13

temporal and atemporal classes. The temporal documents are further classified into three categories: HTS, MTS, and LTS. We extracted several temporal features and selected the top 4 features with high information gain for temporal classification. We compared the results of the proposed classification techniques to four machine learning classifiers: Bayes net, support vector machine, decision tree, and random forest. The results of the classification are presented in terms of precision, recall, F-score, and overall accuracy for each classifier. The results reveal that the rule-based classifier achieves the highest precision and accuracy scores of 0.80 and 0.82, respectively, while the TSS-based classifier achieves a precision score of 0.79 and recall, F-score, and accuracy scores of 0.63, 0.65, and 0.77, respectively. In future, we are interested in determining the focus time of documents belonging to the HTS class using the positions of temporal expressions in the content.

References

[1] Kocabaş İ, Dinçer BTö Karaoğlan B. Investigation of Luhn's claim on information retrieval. Turk J Elec Eng & Comp Sci 2011; 6: 993-1004.

[2] Nittya K, Kjetil N. Using temporal language models for document dating. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases; 7–11 September 2009; Bled, Solvenia. Berlin, Germany: Springer. pp. 338-341.

[3] Niculae V, Zampieri M, Dinu LP, Ciobanu AM. Temporal text ranking and automatic dating of texts. In: Proceedings of the 14th Conference of the European Chapter of the ACL; 26–30 April 2014; Gothenburg, Sweden. pp. 17-21.

[4] Zhang R, Konda Y, Dong A, Kolari P, Chang Y, Zheng Z. Learning recurrent event queries for web search. In: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing; 9–11 October 2010; Massachusetts, USA. pp. 1129-1139.

[5] Metzler D, Jones R, Peng F, Zhang R. Improving search relevance for implicitly temporal queries. In: Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval; 19–23 July 2009; Boston, MA, USA. New York, NY, USA: ACM. pp. 700-7001.

[6] Li X, Croft WB. Time-based language models. In: Proceedings of the Twelfth International Conference on Information and Knowledge Management; 3–8 November 2003; New Orleans, LA, USA. New York, NY, USA: ACM. pp. 469-475.

[7] Costa M, Couto FM, Silva MJ. Learning temporal-dependent ranking models. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval; 6–11 July 2014; Gold Coast, Australia. New York, NY, USA: ACM. pp. 757-766.

[8] Nattiya K. Time-aware approaches to information retrieval. PhD, Norwegian University of Science and Technology, Oslo, Norway, 2012.

[9] Spitz A, Strotgen J, Bogel T, Gertz M. Terms in time and times in context: a graph based term-time ranking model. In: Proceedings of the 24th International Conference on World Wide Web; 18–22 May 2015; Florence, Italy. New York, NY, USA: ACM. pp. 1378-1380.

[10] Strotgen J, Gertz M. Proximity 2-aware ranking for textual, temporal, and geographic queries, In: Proceedings of the 22nd ACM International Conference on Information & Knowledge Management; 27 October–7 November 2013; San Francisco, CA, USA. New York, NY, USA: ACM. pp. 739-744.

[11] Dakka W, Gravano L, Ipeirotis P. Answering general time-sensitive queries. IEEE T Knowl Data Eng 2012; 2: 220-235.

[12] Adam J, Ching M, Katsumi T. Estimating document focus time, In: Proceedings of the 22nd ACM International Conference on Information & Knowledge Management; 27 October–1 November 2013; San Francisco, CA, USA: ACM. pp. 2273-2278.

[13] Adan J, Ching M, Katsumi T. Generic method for detecting focus time of documents. Inf Process Manag 2015; 6: 851-868.

[14] Alonso O, Gertz M, Baeza-Yates R. On the value of temporal information in information retrieval. In: 2007 ACM SIGIR Forum; 2 December 2007. New York, NY, USA: ACM. pp. 35-41.

[15] Omar RA. Temporal Information Retrieval. Davis, CA, USA: University of California, Davis, 2008.

[16] Filannino M, Nenadic G. Mining temporal footprints from Wikipedia. In: Proceedings of the First AHA!-Workshop on Information Discovery in Text; 23 August 2014; Dublin, Ireland. pp. 7-13.

[17] Nattiya K, Kjetil N. Determining time of queries for re-ranking search results. In: International Conference on Theory and Practice of Digital Libraries; 6–20 September 2010; Glasgow, UK. pp. 261-272.

[18] Štajner S, Zampieri M. Stylistic changes for temporal text classification. In: Proceedings of the 16th International Conference on Text, Speech and Dialogue; 1–5 September 2013; Pilsen, Czech Republic. Berlin, Germany: Springer. pp. 519–526.

[19] Fumiyo F, Yoshimi S. Temporal-based feature selection and transfer learning for text categorization. In: Proceedings of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management; 12–14 November 2015; Lisbon, Portugal. pp. 17-26.

[20] Luo X, Zincir-Heywood AN. Analyzing the temporal sequences for text categorization. In: Proceedings of the 8th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems; 20–25 September 2004; Wellington, New Zealand. Berlin, Germany: Springer. pp. 498-505.

[21] Strotgen J, Gertz M. HeidelTime: High quality rule-based extraction and normalization of temporal expressions. In: Proceedings of the 5th International Workshop on Semantic Evaluation; 15–16 July 2010; Los Angeles, CA, USA. New York, NY, USA: ACM. pp. 321-324.

[22] Chang AX, Manning CD. SUTime: A library for recognizing and normalizing time expressions. In: Proceedings of 2012 LREC; 21–27 May 2012; İstanbul, Turkey. pp. 3735-3740.

[23] Fleiss JL. Measuring nominal scale agreement among many raters. Psychol Bull 1971; 76: 378-382.

[24] Xiaoxin Y, Jiawei H. CPAR: Classification based on predictive association rules. In: Proceedings of the 2003 SIAM International Conference on Data Mining; 1–3 May 2003; San Francisco, CA, USA. pp. 331-335.