Article

JIS

Journal of Information Science
1–12

DOI: 10.1177/0165551515620551
jis.sagepub.com
SAGE

TTC-3600: A new benchmark dataset for Turkish text categorization

Deniz Kılınç
Faculty of Technology, Celal Bayar University, Turkey

Akın Özçift
Faculty of Technology, Celal Bayar University, Turkey

Fatma Bozyigit
Faculty of Technology, Celal Bayar University, Turkey

Pelin Yıldırım
Faculty of Technology, Celal Bayar University, Turkey

Fatih Yücalar
Faculty of Technology, Celal Bayar University, Turkey

Emin Borandag
Faculty of Technology, Celal Bayar University, Turkey

Abstract

Owing to the rapid growth of the World Wide Web, the number of documents that can be accessed via the Internet explosively increases with each passing day. Considering news portals in particular, sometimes documents related to categories such as technology, sports and politics seem to be in the wrong category or documents are located in a generic category called others. At this point, text categorization (TC), which is generally addressed as a supervised learning task is needed. Although there are substantial number of studies conducted on TC in other languages, the number of studies conducted in Turkish is very limited owing to the lack of accessibility and usability of datasets created. In this paper, a new dataset named TTC-3600, which can be widely used in studies of TC of Turkish news and articles, is created. TTC-3600 is a well-documented dataset and its file formats are compatible with well-known text mining tools. Five widely used classifiers within the field of TC and two feature selection methods are evaluated on TTC-3600. The experimental results indicate that the best accuracy criterion value 91.03% is obtained with the combination of Random Forest classifier and attribute ranking-based feature selection method in all comparisons performed after pre-processing and feature selection steps. The publicly available TTC-3600 dataset and the experimental results of this study can be utilized in comparative experiments by other researchers.

Keywords

Feature selection; text classification; TTC-3600 dataset; Turkish text categorization

1. Introduction

The rapid growth of the World Wide Web and Internet use is leading to a rapid increase in the amount of unstructured data on the Internet with each passing day. According to the International Data Corporation, the amount of unstructured

Corresponding author:
Deniz Kılınç, Department of Software Engineering, Faculty of Technology, Celal Bayar University, Manisa, Turkey.
Email address: drdenizkilinc@gmail.com; deniz.kilinc@cbu.edu.tr

data on the Internet will exceed up to 40 zettabytes by 2020 and this means that this data will be 50 times larger than the amount of unstructured data on the Internet in 2010.[1] Manual categorization of this unstructured data is almost impossible and there is a need for a continuous automatic categorization process in order to make this data more manageable and reachable. Considering news portals in particular, sometimes documents related to categories such as technology, sports, politics and health seem to be in the wrong category or documents are located in a generic category called 'others'. At this point, approaches and methods in the field of Text Mining (TM) [1], which is an important research area, are needed. The purpose of TM, which is also known as Intelligent Text Analysis, Knowledge Discovery in Text and Text Data Mining in the literature, is extracting valuable and significant information and knowledge from unstructured text documents [2]. TM is an interdisciplinary field that can use machine learning [3], computational linguistics, information retrieval and statistics compositely. One of the most widely utilized methods in TM studies is the method of Text Categorization/Classification (TC) within the supervised-learning category in the field of machine learning. TC creates a model benefiting from a pre-defined set of data and aims to assign uncategorized data into a correct category [4]. In other words, it evaluates uncategorized data based on its content and categorizes it.

One of the most important characteristics of TC is its high dimensionality, in which thousands of features can be generated [5]. Most of the features are irrelevant and result in poor performance of the classifier. Hence, the dimensionality reduction that removes redundant and irrelevant features from dataset before evaluating machine learning algorithms is a critical step in TC. Feature selection is the most widely used dimensionality reduction technique, which selects a relevant subset from the entire features [6].

In this study, a new dataset called TTC-3600, which can be widely used in the studies of TC regarding Turkish news and articles, is created and comprehensive experimental studies are performed on this dataset. Considering the literature, although there are a substantial number of studies conducted on TC in other languages, the number of studies conducted in Turkish is very limited. All TC studies available in Turkish in the literature are investigated within the scope of this study. Since datasets used in other studies are not available or created for different purposes, the dataset used in this study consists of news collected from six news portals and agencies that are very well known in Turkey and this dataset has become publicly available in order to be used in the experimental work of other researchers.

Three different versions of TTC-3600, which are subjected to stemming, are also created and utilized in order to observe the effect of pre-processing on Turkish TC. In the machine learning domain, various types of TC algorithms such as lazy learning, statistical learning and decision tree induction exist. Among these, selection of the best single performing one is a challenging task, as indicated by the No Free Lunch theorem [7]. Based on this theorem, five well-known classifiers Naïve Bayes (NB), Support Vector Machine (SVM), K-Nearest Neighbor (K-NN), Decision Tree (J48) and Random Forest (RF) in the field of TC are evaluated on all versions of the TTC-3600 dataset.

In addition to these experimental studies, impacts of dimensionality reduction methods on Turkish TC are also observed during experimental studies. Correlation-based feature selection (CFS) and attribute ranking-based (ARFS) feature selection methods are employed in order to evaluate the results of dimensionality reduction technique. The experimental results show that the RF classifier is more accurate in all stemming steps (F5, F7 and Zemberek) and feature selection methods applied on the TTC-3600 dataset, and the best ACC result was obtained after applying ARFS on Zemb-DS dataset.

The rest of the paper is organized as follows: the second section offers a comprehensive literature study about TC. In the third section, materials and methods utilized are introduced briefly. Section 4 presents the experimental stud and discusses the experimental results obtained. Finally, the fifth section concludes the paper with some future directions.

2. Related works

Considering the previous studies in the literature, although there are many studies conducted on TC in other languages, the number of TC researches conducted in Turkish is very limited. For instance, there are many TC studies conducted in English, which is one of the most widely spoken languages in the world [8–10]. In addition to this, there has been interesting research in the literature performed in some other languages like Arabic, which has different morphological properties. The study of Ismail et al. [11] aimed to assign articles written in Arabic into the relevant categories. In the study, five well-known algorithms in the field of TC are discussed and the success rates of these algorithms are compared with each other. The other study for Arabic TC proposed by Shaalan and Oudah [12] combines different machine learning algorithms in order to perform named entity recognition. It is claimed that the success rate of the study exceeds 90% and it gives accurate results. Al-Radaideh et al. [13] conducted a study to detect spam emails composed in Arabic. It is claimed that they obtained accurate results from 87% of the messages in the dataset using the Graham statistical filter and rule based filter.

The aim of this study was to investigate Turkish text categorization. In a study conducted by Güran et al. [14], NB, Multinominal Naïve Bayes (MNB), J48 and K-NN TC algorithms were evaluated. Their proposed study was based on the N-gram algorithm. They performed their experimental studies on documents that were either pre-processed or not. According to the results of experimental evaluation, the worst results were obtained when bi-gram, tri-gram representation and K-NN algorithm were performed together. J48 classifier gave the best classification results in general.

In another study conducted by Torunoğlu et al. [15], the importance of pre-processing steps in Turkish TC is observed. Different pre-processing methods and four TC algorithms – NB, MNB, SVM and K-NN – are evaluated. Considering experimental results, it is concluded that the pre-processing step did not show the expected impact on Turkish TC.

Akkus and Ruket [16] suggested that morphological analysis would be a useful method for TC in languages with semantic richness like Turkish. In the study, the contribution of morphological analysis on Turkish TC is studied. First, stems of the words are identified using Fixed Length Stemmer method and K-NN, SVM and NB learning algorithms are evaluated on these stems. Considering the evaluation results conducted on the dataset, using a simple approximation with the first five characters to represent documents instead of results of an expensive morphological analysis gives similar or better results at much lower cost.

In the study of Amasyalı and Beken [17], a different approach regarding TC is presented. They assign words of a text document into a semantic space they have created. They indicate that representing words in semantic space gives better results compared with a bag of words model. According to their experimental results, the Linear Regression Classification Algorithm gives the most successful results.

Amasyalı and Diri [18] propose an n-gram approach to achieve TC for Turkish language in their study. They evaluate NB, SVM, J48 and Random Forrest classification algorithms. As a result of the study, they suggest that classification algorithms conducted with bi-grams give better results compared with classification algorithms conducted with tri-grams. Considering the results of classification algorithms, NB gives more successful results in determining the author of the text, whereas SVM gives more accurate results in terms of determination of genre of the text and gender of the author.

Tüfekçi and Uzun [19] investigate the effect of different term weighting methods to identify the author of the text. In the texts, the different feature vectors of each document are determined by trying different weighting methods after identification of stems of the words. MNB, SVM, Decision Tree and Random Forrest classification algorithms are performed on the vectors created and results are compared with each other. According to the experimental results, the best results are obtained using the SVM algorithm.

In the study of Çataltepe et al. [20], the effect of stem length derived from words in a text on Turkish TC is studied. They obtain short stems from long stems using various methods regardless of the meaning of the words. They aimed to compare accuracy rates formed as a result of classifying vectors weighted using TF $\times$ IDF method and obtained from stems containing fewer characters. As a result, it is observed that Centroid classification method conducted with shortened stems gives better results.

A study conducted by Alparslan et al. [21] aimed to perform information extraction from documents classified within Turkish language. First, they extract word stems using stemming algorithms which are particularly used for Turkish text documents. Document term matrices are formed using the TF $\times$ IDF weighting method with stems obtained after preprocessing. Unlike other studies, SVM and Adaptive Neuro Fuzzy classification algorithms are combined in this study. Considering the experimental results of this method, the method proposed seems to be more accurate.

In the study of Uysal and Gunal [22], it is indicated that pre-processing is important to TC. Emails and news written in both English and Turkish are used as the dataset. The ways in which pre-processing methods affect classification of the text documents are determined. They determine how tokenization, stop-word removal, lower-case conversion and stemming processes and their various combinations affect the accuracy rate of SVM classification algorithms. As a result, it is seen that some pre-processing methods reduce the accuracy rate of classification of text documents, while lower-case conversion and stop-word removal processes improve the accuracy rate of classification of the text documents.

Gunal [23] conducts studies regarding the effect of different feature selection approaches on TC. In these studies, a hybrid selection method is proposed by combining filter and wrapper feature selection methods. According to these studies, features obtained by this method give better results in Turkish TC compared with the single selection method.

There are also some other Turkish text analyses and text retrieval studies other than those performed on Turkish TC. Özalp et al. [24] conducted studies to detect slang words in news and comments made for articles and columns on the Internet. They proposed a study that can automatically filter comments made for online articles, magazines and news texts on the Internet. Unlike most widely used classification studies in the literature, they proposed an irregularity-based approach. This method is suggested to be advantageous in terms of memory management and low counting complexity.

In the study by Özgür et al. [25], an anti-spam filtering method developed for Turkish in particular and specific to agglutinative languages is proposed. The study consists of two separate modules – the Learning Module and the

Morphology Module. In their study, they use both Artificial Neural Network and Bayesian Network algorithms. They claim that they achieve a success rate of 90% in finding spam emails in Turkish on the dataset.

Can et al. [26] hypothesize the items that can affect performance of the text retrieval process and tested validity of these hypotheses one by one. First, they accept the hypothesis suggesting that creating a stop-word list and removing these stop-words will affect the retrieval performance; however, according to the tests conducted, this process does not have a significant impact on the text retrieval process.

Kılıçaslan et al. [27] study anaphora resolution on Turkish texts. They compare different methods to identify pronouns in Turkish texts. In the study, the success of different machine learning algorithms used to analyse Turkish text documents is evaluated. Considering success rates of anaphora resolution, learning models are suggested to be more successful than baselines.

3. Materials and methods applied

3.1. Turkish language overview

Turkish belongs to the Altaic branch of the Ural–Altaic family of languages. The distinctive characteristics of Turkish are vowel harmony and extensive agglutination, which refers to the process of adding suffixes to a stem. It is possible to give the meaning of a sentence in English with only one word in Turkish. For example, the English sentence 'We were not sleeping' is a single word in Turkish: 'sleep' is the stem, and elements meaning 'not', '-ing', 'we' and 'were' are all suffixed to it: 'Uyumuyorduk'. Turkish is derived from the Latin alphabet and consists of eight vowels (a, e, ı, i, o, ö, u, ü) and 21 consonants (b, c, ç, d, f, g, ğ, h, j, k, l, m, n, p, r, s, ş, t, v, y, z); seven of these letters are modified from their original versions in the Latin alphabet (ç, ı, ş, ö, ü, ğ, İ).

3.2. Pre-processing

Pre-processing is one of the most important steps in order to prepare text datasets before TC. Tokenization, stop-word elimination and stemming are the most widely used pre-processing methods. In general, removal-based pre-processing is conducted first. All common separators, operators, punctuations and non-printable characters are removed. Then, stop-word filtering that aims to filter out the most frequent words is performed.

Finally, stemming is applied to obtain the stem of a word that is its morphological root by removing the suffixes that present grammatical or lexical information about the word. The stemming process is based on a hypothesis suggesting that 'words with the same stem are included in relatively similar concepts'. Since Turkish is an agglutinative language and thousands of different words can be derived from a root word, stemming is an important step before performing text categorization. In the present study, a fixed prefix stemming (FPS) [26] approach and a directory-based Turkish stemmer called Zemberek [29] is used. FPS is a pseudo stemming method and it recognizes the first 'n' character in the text and accepts it as the stem. Zemberek is a general-purpose open source NLP toolkit and it includes a suffix dictionary created for stemming.

3.3. Feature representation and weighting

Machine learning classifiers generally handle text documents as bag of words. Vector Space Model is an improved version of bag of words, where each text document is represented as a vector, and each dimension corresponds to a separate term (word) [28]. If a term occurs in the document, then its value becomes non-zero in the vector. When it is considered from a TC perspective, the goal is to construct vectors containing features per category using a training set of the documents. In the Vector Space Model, term weighting is a critical step and three major parts that affect the importance of a term in a text exists as follows: eTerm frequency factor (TF), the inverse document frequency factor (IDF) and document length normalization. The normalization factor is computed as illustrated in equation (1):

$$\sqrt{w_1^2 + w_2^2 + \ldots + w_t^2} \tag{1}$$

where each w equals (TF $\times$ IDF) as in equation (2).

$$w_{ki} = \frac{tf_{ik}\log(N/n_k)}{\sqrt{\sum_{k=1}^{n}(tf_{ik})^2[\log(N/n_k)]^2}} \tag{2}$$

where t_k is the kth term in the document, d_i; tf_{ik} is the frequency of word t_k in document d_i; $\log(N/n_k)$ is the inverse document frequency of word t_k in the dataset; n_k is the number of documents containing the word t_k; and N is the total number of documents in the dataset.

3.4. Text categorization and selected classifiers

As a general description, the aim of TC is classifying uncategorized documents into predefined categories. If we look at it from a machine learning perspective, the aim of TC is to learn classifiers from labelled documents and fulfill classification on unlabelled documents. In the literature, there is a rich collection of machine learning classifiers for TC [4]. The selection of the best performing classifier depends on various parameters, such as thee number of training examples, dimensionality of the feature space, feature independence, over-fitting, simplicity and the system's requirements. Considering the high dimensionality and over-fitting characteristics and related researches conducted on TC, five well-known TC classifiers (NB, SVM, K-NN, J48 and RF) are selected among all TC classifiers. The detailed information about each classifier selected is illustrated in the following section.

3.4.1. Naïve Bayes. The NB classifier is a well-known statistical supervised learning algorithm based on Bayes' Theorem [30]. Conditional probabilities are calculated using all training sets to determine the category in which the text document should be classified. Easy implementation and high performance are important advantages of the NB classifier. Furthermore, it requires a small amount of training data to estimate the parameters and good results are obtained in most cases. Its main disadvantage is that dependencies between features cannot be modelled. NB is frequently applied in the areas of medical diagnosis, TC, pattern recognition and target marketing, and it gives quite successful results. The simple equation of NB classifier is illustrated in equation (3):

$$P(c_j|d) = P(d|c_j)P(c_j)/P(d) \tag{3}$$

where $P(c_j|d)$ is the probability of instance d being in class c_j, $P(d|c_j)$ is the probability of generating instance d given class c_j, $P(c_j)$ is the probability of occurrence of class c_j and $P(d)$ is the probability of instance d occurring.

3.4.2. Support Vector Machine. SVM, which was introduced in 1992, is a classifier based on statistical information theory and structural risk minimization. The SVM algorithm is divided into two algorithms – linear and non-linear SVM. In the linear SVM algorithm, an infinite number of hyper-planes are created in order to separate data and a maximum-margin hyper-plane is selected among all these hyper-planes. Non-linear SVM is used when classes are not linearly separable and data is transferred into a higher dimensional space. In this way, the data becomes linearly separable [31]. The main advantages of SVM are high accuracy and being robust against over-fitting via structural risk minimization by using a regularization parameter. The SVM classifier can also work well with an appropriate kernel even if data is not linearly separable in the base feature space. Memory-intensive performance, hard interpretation and determination of the regularization and kernel parameters and choice of kernel are the disadvantages of SVM.[2] The main applied areas of SVM classifier are TC, pattern recognition, bioinformatics and hand-written character recognition.

3.4.3. K-Nearest Neighbor. K-NN, which has no training phase, is an instance-based lazy learning classification algorithm [32]. According to this algorithm, the categorization process of the document to be categorized is performed by considering at the closest k neighbour among documents that have certain class labels. In the K-NN algorithm, closeness is defined as a similarity measure such as Euclidean distance. Equation (4) calculates the Euclidean distance between two instances $X_1 = (X_{11}, X_{12}, \ldots, X_{1n})$ and $X_2 = (X_{21}, X_{22}, \ldots, X_{2n})$.

$$dist(X_1, X_2) = \sqrt{\sum_{i=1}^{n} (X_{1i} - X_{2i})^2} \tag{4}$$

The implementation of KNN is simple and the cost of the learning phase is zero. It is also robust to noisy data. Despite having these advantages, K-NN also has some drawbacks. Since a description of the learned concepts does not exist, the K-NN model cannot be interpreted and determination of the value of parameter K that specifies the number of nearest neighbours is not easy. Finally, using K-NN it is computationally expensive to find the k nearest neighbours in high dimensions.

3.4.4. J48 Decision Tree. Decision tree learning is a supervised learning method that performs classification process to determine the category of the input document by creating a decision tree over the available training set [33]. In the decision tree created, internal nodes represent attributes of the dataset, branches represent the attribute values and leaves represent the classification label. The J48 classifier is a Java implementation of the C4.5 algorithm, which uses a divide-and-conquer approach for growing the decision tree. J48 is quite successful in the area of TC in particular and has advantages such as having high performance on large datasets and shorter training duration. It builds models that can be easily interpreted and can work with both categorical and continuous values. The main disadvantage of J48 is that small variation in training data may lead to different decision trees.

3.4.5. Random Forest. RF is an ensemble learning method of decision trees proposed by Leo Breiman and Adele Cutler, which grows many classification trees. First, subspaces of features are randomly selected to construct branches of decision trees [34]. Then, training data is created to be used to generate each individual tree. Finally, an RF classification model is created by combining all individual trees. All input parameters are passed to each individual tree in the forest for categorization process of a document. Classification label returns are collected from all trees in the forest and the label with highest vote is selected as the predicted outcome.

RF is a highly accurate classifier that runs efficiently on large datasets and can handle thousands of input features without any deletion. It has an effective method for estimating missing data and maintains accuracy when a large proportion of the data is missing. RF also contains an experimental method for detecting feature interactions. RF classifier may not run effectively in a dataset including categorical variables with varying numbers of levels, because random forests are biased in favour of those attributes with more levels.

3.5. Feature selection

In a text classification approach, if too many features exist in a dataset, it may result in over-fitting and accuracy of the classifier will presumably decrease. In addition, as the number of features increases, performing classification becomes impossible because of the lack of computational resources. Consequently, it is important to remove redundant and irrelevant features from the dataset before evaluating machine learning algorithms [33]. Feature selection is an important step to reduce dimensionality and remove irrelevant features. Feature selection methods are categorized as filter-based and wrapper-based methods. Filter-based methods are based on specific characteristics of the training instances for selecting some features without applying any learning algorithm. On the other hand, wrapper-based methods attempt to find features better suited to a pre-defined learning algorithm or classifier. In a classification task, which has a high dimensionality characteristic, the filter-based methods are usually selected because of their computational efficiency. Therefore, two well-known filter-based feature selection approaches are utilized in this research and details of these approaches are presented in the remaining part of the section.

3.5.1. Correlation-based feature selection. The CFS is a filter-based feature selection method used for evaluating subsets of features on the basis of the simple idea: 'Good feature subsets contain features that are highly correlated with the classification, but contrarily have low correlation with other features' [35]. Equation (5) calculates the merit of a feature subset S including k features:

$$Merit_{Sk} = \frac{k\overline{r_{cf}}}{\sqrt{\mathrm{k} + \mathrm{k}(\mathrm{k}-1)\overline{r_{ff}}}} \tag{5}$$

where, $\overline{r_{cf}}$ is the average value of all feature-classification correlations, and r_{ff} is the average value of all feature–feature correlations. The CFS method is usually utilized with a heuristic search strategy such as best-first search, greedy stepwise and genetic search.

3.5.2. Attribute ranking-based feature selection. The idea behind ARFS is separately ranking features according to their predictive capabilities for the category and selecting the top ranking ones. One of the most widely used methods in the area of machine learning is the information gain-based method [36]. For each category and feature, an information gain score to be used in the ranking process is calculated and the top N ranking features are selected as the feature subset. The information gain (IG) of the feature f_k over the class c_i is calculated by equation (6):

$$\mathrm{IG}(f_k, c_i) = \sum_{c \in \{c_i, c_i\}} \sum_{t \in \{f_k, f_k\}} P(t, c) \log \frac{P(f, c)}{P(f)P(c)} \quad (6)$$

where $P(c)$ is the proportion of the documents in category c over the total number of documents and $P(f, c)$ is the proportion of documents in the category c that contain the feature f over the total number of documents. $P(f)$ is the proportion of the documents containing the feature f over the total number of documents [37].

4. Experimental study

In this section, we present detailed information about the experimental procedure applied, the TTC-3600 dataset created and conducted, the performance evaluation criteria considered and the experimental results obtained.

4.1. Experimental method

The experiments are performed using the implementations of NB, J48, RF, SVM and K-NN classifiers in the WEKA (Waikato Environment for Knowledge Analysis) version 3.6.12 [38]. In this study, the default parameters are set for each WEKA classifier implemented and feature selection method since these parameters give promising experimental results [2]. For NB with continuous variables, no kernel method for estimation of the distribution is used. For SVM, a non-linear kernel of degree 3 with WEKA's default settings is utilized. Default RF parameters 100 and 1 are selected, where the first number is the number of trees and the second number is the random number seed used for each tree. Furthermore, the default K-NN and J48 classifier parameters are employed in the research. For K-NN, the value of parameter k is selected as 1, distance weighting is not applied and Euclidean distance is selected as distance function.

Each classifier is tested with 10-fold cross-validation, which is a common strategy for classifier performance estimation. In this strategy, each dataset is split into 10 blocks. One single block is retained as the validation data for testing the model and the remaining $k - 1$ blocks are used as training data. The cross-validation process is then repeated 10 times.

In this study, two FS methods – CFS and ARFS – are used in order to evaluate the performance of feature selection methods applied on the TTC-3600 dataset. For CFS, the CfsSubsetEval evaluator of WEKA data mining tool with BestFirst search strategy is used to select the best feature subset. For ARFS, InfoGainAttributeEval evaluator with Ranker search method is utilized to rank features in accordance with their information gain score. Instead of empirically selecting N features with the highest ranking score, all features that are in keeping with an information gain score higher than 0 are selected by setting the value of threshold parameter of the Ranker search method to zero.

4.2. TTC-3600 dataset

Since datasets used in other studies are either not accessible or created for different purposes, a new dataset called TTC-3600 is created. The most important feature of this dataset, which can be widely used in the studies of TC regarding Turkish news and articles, is being simple to use and well documented. The dataset consists of a total of 3600 documents including 600 news/texts from six categories – economy, culture-arts, health, politics, sports and technology – obtained from six well-known news portals and agencies (Hürriyet,[3] Posta,[4] İha,[5] HaberTürk,[6] Radikal[7] and Zaman[8]). Documents of TTC-3600 dataset were collected between May and July 2015 via Rich Site Summary (RSS) feeds from six categories of the respective portals. A special RSS Feeder, which allows collection of XML-Format RSS Feeds from any portal, was developed using C# programming language on Visual Studio 2013 IDE to fetch the RSS feeds. In the study, $<\text{title}>$ and $<\text{description}>$ XML elements of RSS feeds are taken into consideration for text categorization. Since these items contain unnecessary data for TC, removal-based pre-processing is conducted. All java scripts, HTML tags ($<\text{img}>$, $<\text{a}>$, $<\text{p}>$, $<\text{strong}>$ etc.), operators, punctuations, non-printable characters and irrelevant data such as advertising are removed.

Three additional dataset versions are created on TTC-3600 by implementing different stemming methods. In all versions of datasets, first, removal-based pre-processing, which is explained in Section 3.2 in detail, is used. Then Turkish stop-words that have no discriminatory power (pronouns, prepositions, conjunctions, etc.) in regard to TC are removed from datasets except for the original one. In this study, a semi-automatically constructed stop-words list [26] that contains 147 words is utilized.

After completing the process of pre-processing, text documents containing stems in all versions of datasets are transformed into a document–term matrix by utilizing text2arff tool [39], which is a feature extraction software, using the TF $\times$ IDF weighting scheme. Then, each matrix belonging to dataset versions is converted into attribute relation file format (ARFF), which is the proper format for WEKA to be executed.

Table 1. TTC-3600 dataset versions.

No.	Dataset name	Stop word filtering	Stemmer	Number of documents	Number of features
1	Original-DS	No	No stemmer	3600	7508
2	F5-DS	Yes	FPS-5	3600	3209
3	F7-DS	Yes	FPS-7	3600	4814
4	Zemb-DS	Yes	Zemberek	3600	5693

Table 1 gives information about the TTC-3600 dataset version. In datasets F5-DS and F7-DS, stemming is performed using an FPS approach and the first five and seven characters of the words are selected as the stem, respectively. In the Zemb-DS dataset, the Zemberek NLP toolkit is used as the stemmer. In Original-DS, F5-DS, F7-DS and Zemb-DS datasets, there are 7508, 3209, 4814 and 5693 words (features), respectively.

The dataset and files have been made publicly available in order to have repeatable results for experimental evaluation on TTC-3600 dataset.[9] Each version of the TTC-3600 dataset includes two types of files in addition to original text files that are pre-processed. The first file with a '.txt' extension contains the names and ids of the features, whereas the second file in ARFF format describes a list of instances sharing a set of features.

4.3. Evaluation criteria

In machine learning domain, there are different evaluation criteria used to evaluate classifiers. All criteria are generated from a confusion matrix [40], which contains actual and predicted classification information. True positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) denote the four different prediction outcomes. In this study, the most accepted evaluation criterion, ACC, is utilized. Each criterion is described in the following.

Accuracy (ACC) is the most widely used performance evaluation criterion, which is the ratio of the total number of class files that are classified correctly. It is calculated by using equation (7):

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{7}$$

Precision is the proportion of correctly classified class files with faults. Recall is the proportion of correctly classified class files with faults. Precision and Recall are calculated using equations (8) and (9), respectively:

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{8}$$

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{9}$$

4.4. Experimental results and discussion

Figure 1 presents ACC evaluation criteria results of all classifiers on the TTC-3600 dataset. The aim of experimental studies performed to form this figure is to evaluate the performance of TC classifiers on dataset versions created using different stemming methods. Considering the experimental results, RF is evaluated as the most accurate classifier in terms of ACC. In addition, the highest ACC value is achieved by the RF classifier in all datasets regardless of stemming. ACC values obtained by this classifier are 88.6, 87.9, 88.3 and 90.1% for Original-DS, F5-DS, F7-DS and Zemb-DS datasets, respectively.

On the other hand, K-NN has the lowest ACC values among all classifiers and its ACC results are $< 60\%$. The closest criteria results to RF are achieved by SVM (except Zemb-DS), which is a kernel-based classifier. NB classifier gives more accurate criteria results in the Zemb-DS dataset compared with SVM.

According to the data presented in Figure 1, there is a 3% ACC difference maximum between evaluation criteria results of classifiers used in the study on the Original-DS dataset and evaluation criteria results on three datasets created after stemming. Considering that the Original-DS, F5-DS, F7-DS and Zemb-DS datasets have 7508, 3209, 4814 and 5693 features, respectively, the number of features is reduced dramatically; however, the effect of this reduction on ACC is found to be 3% maximum. This situation indicates that the accuracy effect of pre-processing on experimental results conducted on Turkish texts before TC is not promising.

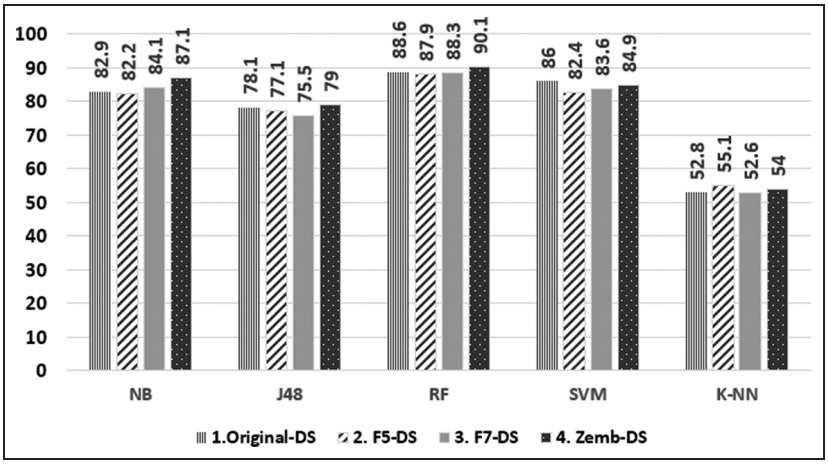

Figure 1. Experimental ACC percentage results of classifiers on datasets.

Table 2. The number of remaining features after feature selection methods.

No.	Without FS	CFS	ARFS
Original-DS	7508	55	1684
F5-DS	3209	35	942
F7-DS	4818	63	1241
Zemb-DS	5693	52	1551

Table 3. The effect of FS methods on the experimental results.

	ACC without FS				ACC of CFS				ACC of ARFS			
	OrgDS	ZembDS	F5DS	F7DS	OrgDS	ZembDS	F5DS	F7DS	OrgDS	ZembDS	F5DS	F7DS
NB	82.94	87.17	82.22	84.03	78.97	80.44	75.25	78.56	82.94	87.19	82.22	84.06
J48	78.06	79.00	77.14	75.50	76.72	78.19	71.67	74.78	78.97	79.39	77.36	75.97
RF	88.53	**90.10**	87.92	88.25	80.17	**81.42**	75.44	78.67	88.87	**91.03**	88.28	88.59
SVM	86.03	84.97	82.39	83.56	69.31	69.61	68.17	69.19	79.53	76.86	74.97	76.92
KNN	52.83	54.00	55.11	52.67	73.11	74.97	69.44	72.56	64.44	65.25	64.33	62.56

Generally, if we evaluate the success of the stemming methods, classifier evaluation results obtained from F5-DS and F7-DS datasets, which are subjected to stemming by FPS approach, are worse than classifier evaluation results obtained from the original datasets (except K-NN). Classifier results on the Zemb-DS dataset that is created by using Zemberek NLP toolkit are better than the results obtained from original dataset (except SVM). As a result, in all TTC-3600 datasets, the stemming process performed using Zemberek classifier outperforms all other methods.

In this study, after CFS and ARFS methods are performed in order to observe the effect of FS methods, the remaining numbers of features for each dataset are presented in Table 2. As a result of the CFS method, which is combined with a heuristic search strategy best-first search, it is eliminated since about 85–90% of the features in the datasets are found to be irrelevant.

On the other hand, the number of remaining features is greater than the number of features obtained by CFS after applying ARFS method. For example, there are 4818 features in the initial stage of F7-DS dataset, whereas the number of remaining features at the end of CFS process is 63 and this value is 1241 at the end of ARFS.

Table 3 shows the performance comparison of feature-selection methods in terms of ACC on four datasets. As it can be seen from Table 3, the ACC performance of the all classifiers except K-NN on all datasets is reduced after applying CFS method. For example, before the process, ACC values in OrgDS, ZembDS, F5DS and F7DS for NB classifier were

82.94, 87.17, 82.22 and 84.03, respectively; however, they became 78.97, 80.44, 75.25 and 78.56 after performing CFS, respectively. A similar decrease is also observed for RF classifier before and after CFS. For J48, since the values of ACC results are already low, a decrease at minimum level occurs after CFS.

One of the highest amount of performance decreases is observed in a non-linear kernel-based SVM classifier. When SVM is performed after applying CFS, the decrease observed in the performance of ACC values is around 12–15%. SVM classifier, which is one of the state-of-the-art algorithms of today, harnesses the problem of over-fitting via structural risk minimization by supporting regularization. Any attempt, such as discretization or feature selection, will invalidate the bounds on performance and potentially overwhelm the structural risk minimization principle. Re-considering the results given in Figure 1, SVM is the only classifier in which the ACC criterion value is not increased in the ZembDS dataset. As a result, SVM has a fairly robust algorithmic design against uninformative features and produces better results when no selection or reduction is performed.

After performing CFS, the only classifier with significantly increased performance is K-NN. For example, the value of ACC in the OrgDS dataset is increased by about 21% from 52.83 to 73.11%. The main reason for this situation is that K-NN algorithm is directly affected by a phenomenon called 'curse of dimensionality' [41] in the literature in an environment that has high-dimensional properties. More specifically, in high dimensions, Euclidean distance is ineffective since all vectors are almost at equal distances to the search query vector.[10] Since the number of features is significantly decreased after performing CFS, the K-NN algorithm has a much more accurate performance compared with its state with no feature selection.

Consequently, since around 85–90% of the features in TTC-3600 datasets are eliminated after performing CFS and there are also discriminative features eliminated for categories, ACC performance values of the classifiers except K-NN are decreased. In addition, it is observed that feature selection implementation of SVM classifier, which is a non-linear kernel-based classifier that can work better in a high-dimensional environment, reduces the accuracy.

In addition to the CFS results, considering the ARFS results given in Table 3, after performing ARFS, NB, J48 and RF classifiers obtained either similar or better ACC results compared with the values obtained from original datasets. For example, the best ACC value in this study (91.03%) is obtained by performing RF classifier on ZembDS dataset after applying ARFS. In addition, the ACC values obtained by performing these three classifiers after performing ARFS are more successful compared with the implementation of CFS.

The SVM classifier has a worse performance (7–8%) compared with ACC values in the original dataset after performing ARFS; on the other hand, it gives much more accurate results and higher ACC values compared with the results obtained by CFS. This result obtained in the TTC-3600 dataset is not surprising when the high performance of SVM in a non-linear-based and high-dimensional environment is taken into consideration. Because the number of features remaining after performing ARFS is much greater than the number of features remaining after performing CFS and SVM, that shows better performance in a high-dimensional environment.

ACC values of K-NN classifier are increased by about 8–12% in other datasets compared with its values on the original dataset after applying ARFS; however, its ACC values are decreased compared with CFS. It can be concluded from the feature selection experiments that the performance of ARFS is superior to that of CFS except for K-NN, which showed a great success with a 74.97% ACC value in ZembDS dataset using only 52 features remaining after performing CFS. Accordingly, it can be speculated that, even though the ACC results of K-NN classifier decrease in a high-dimensional environment, it can be promising when performed with a small number of features or in other words, when different dimensionality reduction methods are applied.

Finally, the RF classifier is more accurate in all stemming steps (F5, F7, Zemberek) applied in the TTC-3600 dataset and feature selection methods (CFS, ARFS) and the best ACC result is obtained in the ZembDS dataset after applying ARFS.

5. Conclusion and future works

In this study, intensive experimental studies on Turkish TC, which are very limited compared with other languages, are employed and all accessible research in the literature is discussed. A new dataset called TTC-3600, which can be widely used in TC studies regarding Turkish news and articles, has been created by collecting news from six well-known news portals and agencies in Turkey and made publicly available in order to be used in comparative experiments by other researchers. Three different versions of TTC-3600 dataset, which are pre-processed (stemming, stop-word elimination, etc.) and can be used in TC studies regarding Turkish news and articles, have also been created in addition to the original dataset and used in the experiments of the study. Detailed information about TTC-3600 dataset is presented in Section 4.2.

Five well-known classifiers – NB, SVM, K-NN, J48 and RF – within the field of TC are evaluated on the TTC-3600 dataset. In addition, CFS and ARFS feature selection methods are also utilized in order to observe the impacts of feature selection methods on Turkish TC. The experimental results indicate that in all comparisons performed after pre-processing and feature selection steps, the RF classifier gives more accurate results and the best ACC value of 91.03% is obtained in the dataset version of Zemb-DS after applying ARFS.

In future studies, other TC classifiers, ensemble learning methods, different types of feature selection approaches and n-gram-based dimensionality reduction method can be used in order to perform research on the TTC-3600 dataset in more detail. Other future work is constructing a new big data set by collecting many more documents and investigating horizontally scaled TC by utilizing a library like Hadoop MapReduce [42].

Funding

This research received no specific grant from any funding agency in the public, commercial or not-for-profit sectors.

Notes

1. https://en.wikipedia.org/wiki/Unstructured_data.
2. http://axon.cs.byu.edu/Dan/678/miscellaneous/SVM.example.pdf.
3. http://dosyalar.hurriyet.com.tr/rss.
4. http://www.posta.com.tr/rss.
5. http://www.iha.com.tr/rss.html.
6. http://www.haberturk.com/rss.
7. http://www.radikal.com.tr/rss.
8. http://www.zaman.com.tr/rss_rssMainPage.action?sectionId=341.
9. https://github.com/GitCBU/TTC-3600.
10. https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm.

References

[1] Chen SY and Liu X. The contribution of data mining to information science. *Journal of Information Science* 2004; 30(6): 550–558.
[2] Amancio DR et al. A systematic comparison of supervised classifiers. *PloS One* 2014; 9(4): 94–137.
[3] Michie D, Spiegelhalter DJ and Taylor CC. *Machine learning, neural and statistical classification*. New York: Ellis Horwood Limited, 1994.
[4] Sebastiani F. Machine learning in automated text categorization. *ACM Computing Surveys* 2002; 34(5): 1–47.
[5] Jieming Y, Zhaoyang Q and Liu Z. Improved feature-selection method considering the imbalance problem in text categorization. *The Scientific World Journal* 2014; doi:10.1155/2014/625342.
[6] Onan A. Classifier and feature set ensembles for web page classification. *Journal of Information Science* 2015; doi: 10.1177/0165551515591724.
[7] Wolpert DH and Macready WG. No free lunch theorem for search. Technical Report SFI-TR-05–010, Santa Fe Institute, 1995.
[8] Read J. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In: *Proceedings of the ACL student research workshop*, 2005, pp. 43–48.
[9] Zhang P and He Z. Using data-driven feature enrichment of text representation and ensemble technique for sentence-level polarity classification. *Journal of Information Science* 2015; doi: 10.1177/0165551515585264.
[10] Cavnar WB and Trenkle JM. N-gram based text categorization. In: *Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval*, 1994, pp. 161–175.
[11] Ismail H et al. Automatic Arabic text categorization: A comprehensive comparative study. *Journal of Information Science* 2015; 41(1): 114–124.
[12] Shaalan K and Oudah M. A hybrid approach to Arabic named entity recognition. *Journal of Information Science* 2014; 40(1): 67–87.
[13] Al-Radaideh QA, AlEroud AF and Al-Shawakfa EM. A hybrid approach to detecting alerts in Arabic e-mail messages. *Journal of Information Science* 2012; 38(1): 87–99.
[14] Güran A, Akyokuş S, Güler N and Gürbüz Z. Turkish text categorization using n-gram words In: *Proceedings of the international symposium on innovations in intelligent systems and applications (INISTA)*, 2009, pp. 369–373.
[15] Torunoğlu D, Çakırman E, Ganiz MC et al. Analysis of preprocessing methods on classification of Turkish texts. In: *Proceedings of international symposium on innovations in intelligent systems and applications*, 2011, pp. 112–118.
[16] Akkus BK and Ruket C. Categorization of Turkish news documents with morphological analysis. In: *Proceedings of the ACL student research workshop*, 2013, pp. 1–8.

[17] Amasyalı MF and Beken A. Measurement of Turkish word semantic similarity and text categorization application. In: *Proceedings of IEEE signal processing and communications applications conference*, Antalya, Turkey, 9–11 April 2009. New York: IEEE, pp. 1–4.
[18] Amasyalı MF and Diri B. Automatic Turkish text categorization in terms of author, genre and gender. In: *Natural language processing and information systems*. Berlin: Springer, 2006, pp. 221–226.
[19] Tüfekçi P and Uzun E. Author detection by using different term weighting schemes. In: *Proceedings of IEEE signal processing and communications applications conference (SIU)*, Trabzon, Turkey, 24–26 April 2013. New York: IEEE, pp. 1–4.
[20] Çataltepe Z, Turan Y and Kesgin F. Turkish document classification using shorter roots. In: *Proceedings of IEEE signal processing and communications applications conference (SIU)*, Eskisehir, Turkey, 11–13 June 2007. New York: IEEE, pp. 1–4.
[21] Alparslan E, Karahoca A and Bahşi H. Classification of confidential documents by using adaptive neurofuzzy inference systems. *Procedia Computer Science* 2011; 3: 1412–1417.
[22] Uysal AK and Gunal S. The impact of preprocessing on text classification. *Information Processing and Management* 2014; 50: 104–112.
[23] Gunal S. Hybrid feature selection for text classification. *Turkish Journal of Electrical Engineering and Computer Sciences* 2012; 20: 1296–1311.
[24] Özalp N, Yılmaz G and Ayan U. Novel comment filtering approach based on outlier on streaming data. In: *Proceedings of IEEE signal processing and communications applications conference (SIU)*, Mugla, Turkey, 18–20 April 2012. New York: IEEE, pp. 1–4.
[25] Özgür L, Güngör T and Gürgen F. Adaptive anti-spam filtering for agglutinative languages. *Pattern Recognition Letter*s 2004; 25(16): 1819–1831.
[26] Can F, Kocberber S, Balcik E, Kaynak C, Ocalan HC and Vursavas OM. Information retrieval on Turkish texts. *Journal of the American Society for Information Science and Technology* 2008; 59(3): 407–421.
[27] Kılıçaslan Y, Güner ES and Yıldırım S. Learning-based pronoun resolution for Turkish with a comparative evaluation. *Computer Speech and Language* 2009; 23: 311–331.
[28] Salton G and Christopher B. Term-weighting approaches in automatic text retrieval. *Information Processing and Management* 1988; 24: 513–523.
[29] Akin AA and Akin MD. Zemberek, an open source NLP framework for Turkic Languages. *Structure* 2007; 10: 1–5.
[30] Yildirim P and Birant D. Naive Bayes classifier for continuous variables using novel method (NBC4D) and distributions. In: *Proceedings of IEEE international symposium on innovations in intelligent systems and applications (INISTA) proceedings*, Alberobello, Italy 23–25 June 2014. New York: IEEE, pp. 110–115.
[31] Sebastiani F. Text categorization. In: *Proceedings of Text Mining and its Applications to Intelligence*, CRM and Knowledge Management, Southampton, UK, 2005. WIT Press, pp.109–129.
[32] Aha DW, Kibler D and Albert MK. Instance-based learning algorithms. *Machine Learning* 1991; 6(1): 37–66.
[33] Quinlan JR. C4.5: Programs for machine learning. *Machine Learning* 1993; 16(3): 235–240.
[34] Xu B, Guo X, Ye Y and Cheng J. An improved random forest classifier for text categorization. *Journal of Computers* 2012; 7(12): 2913–2920.
[35] Hall M. Correlation-based feature selection for machine learning. *PhD thesis, Department of Computer Science, University of Waikato, New Zealand*, 1999, pp. 51–74.
[36] Yang Y and Pedersen JO. A comparative study on feature selection in text categorization. In: *Proceedings of the 14th international conference on machine learning (ICML '97)*, Nashville, TN, 1997. San Francisco, CA: Morgan Kaufmann, pp. 412–420.
[37] Youn E and Jeong MK. Class dependent feature scaling method using naive Bayes classifier for text datamining. *Pattern Recognition Letters* 2009; 30(5): 477–485.
[38] Witten IH and Frank E. *Data mining: Practical machine learning tools and techniques*. San Francisco, CA: Morgan Kaufman, 2005.
[39] Amasyalı M et al. Text2arff: Automatic feature extraction software for Turkish texts. In: *Signal Processing and communications applications conference (SIU)*, Diyarbakir, 22–24 April 2010. New York: IEEE, pp. 629–632.
[40] Kohavi R and Provost F. On applied research in machine learning. *Machine Learning* 1998; 30(2–3): 127–132.
[41] Kevin B et al. When is 'nearest neighbor' meaningful. In: *7th International conference on database theory*, Jerusalem, 10–12 January 1999. New York: IEEE, pp. 217–235.
[42] Meijing L, Xiuming Y and Ryu KH. MapReduce-based web mining for prediction of web-user navigation. *Journal of Information Science* 2014; 40(5): 557–567.