# Developing a Text Categorization Template for Turkish News Portals

Cagri Toraman, Fazlı Can, Seyit Koçberber
ctoraman, canf {@cs.bilkent.edu.tr}; seyit@bilkent.edu.tr
Bilkent Information Retrieval Group
Computer Engineering Department
Bilkent University, 06800, Ankara, Turkey

*Abstract*—**In news portals, text category information is needed for news presentation. However, for many news stories the category information is unavailable, incorrectly assigned or too generic. This makes the text categorization a necessary tool for news portals. Automated text categorization (ATC) is a multifaceted difficult process that involves decisions regarding tuning of several parameters, term weighting, word stemming, word stopping, and feature selection. In this study we aim to find a categorization setup that will provide highly accurate results in ATC for Turkish news portals. We also examine some other aspects such as the effects of training dataset set size and robustness issues. Two Turkish test collections with different characteristics are created using Bilkent News Portal. Experiments are conducted with four classification methods: C4.5, KNN, Naive Bayes, and SVM (using polynomial and rbf kernels). Our results recommends a text categorization template for Turkish news portals and provides some future research pointers.**

***Keywords-text categorization; news portals; Turkish news***

## I. Introduction

It is easy to reach news from various resources like news portals today. In news portals news categorization makes the news articles more accessible. Manual news categorization is slow, expensive and inconsistent [1]. Therefore ATC is one of the primary tools of news portal construction. Bilkent News Portal (http://139.179.21.201/PortalTest/) is a typical news portal system that displays numerous news articles coming from several RSS resources. It has been active since 2008 and provides links to more than 1.5 million news articles. In such portals, news articles coming from RSS resources include category tags; however, in several cases these tags are empty , incorrect, or too generic. For example “last minute” is used very frequently as a news category. Therefore it may be even better to apply an automatic categorization to all incoming articles.

There are several classification methods in the literature. Applying ATC is a complex process. Their success varies according to decisions regarding different aspects of text categorization such as parameter tuning, term weighting, preprocessing in terms of word stemming and word stopping, and feature selection. Since there are various resources feeding news portals in long periods and number of aggregated news changes according to recent news agenda it is important to choose a proper training set size for ATC. Furthermore, training data should be a good representative of the recent news agenda. In practice it will be automatically created from the tagged current news articles received from reliable news resources. Training with too many or too few most recent news stories can affect the categorization process in a negative way since both cases misrepresent the current news agenda. Therefore, it is important to have an accurate categorization template for effective results. (In the paper "news", "news article", "news story" and "document" are used interchangeably.) All of these issues motivate us to examine the effects of such decisions on automated classification of news articles. Furthermore, we want to investigate these issues within the content of Turkish news portals.

In literature, there are various studies regarding automated text categorization [2, 3]. Studies on Turkish text categorization is limited. The work reported in [4] analyzes text categorization methods in Turkish texts to see the effect of n-gram models. Another work [5] uses a similar approach for author, genre and gender classification. The authors in [6] consider some aspects of news categorization with a small dataset.

The contributions of this paper are that we recommend a comprehensive ATC template for Turkish news articles and examine impacts of ATC-issues on news portals. In Section 2, we explain our categorization template in details. How we setup our experiments is explained in Section 3. The experiment results are given in Section 4. Finally, Section 5 concludes the paper and provides some future research pointers.

## II. Developing a Template

Our template for Turkish news articles consists of two main parts: (i) determining a highly accurate categorization setup for Turkish news articles that will provide highly accurate results and (ii) examining issues on news portals.

Before going into news portal issues, it is important to see how Turkish language reacts to techniques used in text categorization. In this respect, we aim to find an highly accurate setup including various aspects used in text categorization.

Firstly, different types of machine learning-based classifiers result in different results. We choose to use C4.5, KNN (k-

**978-1-61284-922-5/11/$26.00 ©2011 IEEE**

Nearest Neighbor), Naive Bayes (NB) and SVM (Support Vector Machines) with the kernels polynomial(poly) and rbf. KNN [7] has been researched over years and becomes a traditional benchmark. SVM [8] becomes popular in recent years, since it is reported to give good results. There are some modified versions of SVM that are faster than the traditional one. One of them, SMO (Sequential Minimal Optimization) [9] is used in this study. C4.5 [10] which is a decision tree approach and probability-based Naive Bayes [11] are another popular classification approaches studied in literature.

Classification methods usually have parameters giving different results with respect to the given data. KNN needs k value representing number of nearest neighbors. Choosing an optimal k value is impossible due to the variations between data sets. SVMs are linear classifiers in their simple form; but they can also learn non-linear classifiers by using kernel functions like poly or rbf [12]. These kernels vary with degree and width parameters respectively. Lastly, C4.5 decides to prune by looking a threshold called confidence value.

Term weighting techniques are important in information retrieval literature. In its simple form, terms are weighted as binary – 0s or 1s with respect to their occurrence. Term Frequency (*tf*) takes how many times a term appears in document into account. Lastly, *tf.idf* [13], which is a traditional approach in IR, uses occurence of a term in other documents as well as term frequency.

Preprocessing techniques include using stemmers and applying a stopwordlist which removes frequently used words in that language. Using stems of words reduces the dimensionality of the given data. There are various studies to develop stemming algorithms in English like [14]. In Turkish, we choose to apply F*n* stemming approach which simply uses first *n* characters of a word. We use the Turkish stopwordlist given in [15].

Feature selection is used in text categorization to choose the most discriminating features. Feature means either a word or a phrase. We use its simple form as a word. Features are obtained by calculating a scoring function. We choose to apply information gain, gain ratio, chi-squared statistic and relieff [16, 17, 18].

We aim to obtain a highly accurate ATC setup for Turkish news articles by investigating the effects of: (i) parameter tuning, (ii) term weighting , (iii) stemming and stopping (that we also refer to as preprocessing), and (iv) indexing (feature selection). News portals get news articles from various news resources and these documents accumulate with time. In news portals, we observe that:

(i) It is important to decide how many of the incoming articles should be used during training. Choosing an appropriate training size for all applications is a common concern [19].

(ii) Content of news articles changes with time. Content analysis is an old research topic. A robust classifier in our study is expected to have small differences in its performance as news stories changes with time.

TABLE I. CATEGORY INFORMATION OF BILCAT-MIL

| Category | # Train Docs | # Test Docs |
|---|---|---|
| Sports | 572 | 258 |
| Economy | 472 | 208 |
| Turkey | 458 | 199 |
| World | 411 | 168 |
| Politics | 397 | 185 |
| Columnists | 357 | 201 |

TABLE II. CATEGORY INFORMATION OF BILCAT-TRT

| Category | # Train Docs | # Test Docs |
|---|---|---|
| Sports | 716 | 337 |
| World | 580 | 292 |
| Turkey | 473 | 252 |
| Economy | 368 | 190 |
| Health | 165 | 61 |
| Culture&Art | 140 | 80 |

## III. EXPERIMENTAL DESIGN

### A. *Dataset Description*

Since our concern is on Turkish news articles, data used in experiments should be in Turkish. We created two different data sets called BilCat-MIL and BilCat-TRT by exploiting Bilkent News Portal. Categories of these data are assigned by RSS resources. These datasets can be accessed at (http://cs.bilkent.edu.tr/~ctoraman/datasets).

Category information of BilCat-MIL and BilCat-TRT are given in Table I and Table II respectively. BilCat-MIL and BilCat-TRT consist of 3,886 and 3,654 documents coming from Milliyet and TRT that are collected between 01.11.2010 – 26.11.2010 and 01.01.2011 – 25.02.2011 respectively. They respectively contain 50,048 and 52,042 unique words.

BilCat-MIL is deliberately chosen to be more balanced than BilCat-TRT to observe if results differ. We divide our data sets such that train data are approximately two times of test data to provide sufficient sizes for both sets. We do not use k-fold cross-validation or random sampling procedures since content of news articles changes as time passes: old documents must be used for training and new documents must be used for testing (but not the other way). They also violate our training set size and time distance experiments. The details of our experimental procedures are explained in the next section.

### B. *Template Development Procedure*

The algorithms experimented in this study are conducted with the help of Weka [20]. The most frequent 1,000 unique words per category is used to avoid overfitting [12] to increase efficiency. In the first part of our template development, experiments are based on iterative optimization, a technique similar to game theory [21]. In the first iteration, default parameters are selected and the best parameters are obtained through four setup levels. The following iterations start with parameters that are selected at the end of the previous iteration. We stop iterations heuristically when accuracy difference between two iterations is less than 0.5%. Parameters at the end of the last iteration construct a highly accurate setup. Each iteration consists of five setup-levels:

i. *setup-0 (default)*: In the beginning of the first iteration, parameters of all classifiers are adjusted to their default values. Binary term weighting is used. Preprocessing and feature selection are not applied. The following iterations start with parameters that are selected at the end of the previous iteration.

ii. *setup-1*: Parameter of a classifier is to be determined. The other parameters are the same as parameters obtained at the end of the previous iteration (if any, otherwise *default-setup*) - the same approach is applied in the following setup level as well -.

iii. *setup-2*: The term weighting scheme of a classifier is to be determined. The classifier parameters are fixed as determined by *setup-1*.

iv. *setup-3*: The effect of preprocessing is to be determined. The classification parameters and term weighting settings are the same as determined by *setup-1* and *setup-2*, respectively.

v. *setup-4*: The effect of feature selection is to be determined. The classification parameters, term weighting, and preprocessing settings are the same as determined by *setup-1*, *setup-2* and *setup-3* respectively.

In the second part of our template development, we examine different training set sizes and different time distances between training and test sets. While examining training set size, we choose sub-datasets of different size with different time spans all ending at the beginning time of test set (see Fig. 1-a). By making training documents adjacented to test documents, we make sure that training set reflects the recent news content. While examining different time distances between training and test sets, we choose sub-datasets of same size (see Fig. 1-b). By keeping the size of training sub-datasets the same, we make sure that we eliminate the effect of different training set sizes. By this way, we examine the effect of the time distance between train and test sets.

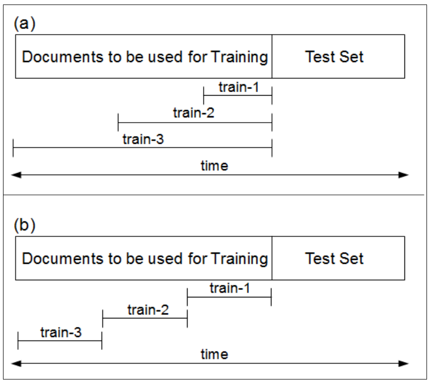

Figure 1. Development procedure for the second part of text categorization template: Analyzing (a) effect of training set size, (b) effect of time distance between training and test sets. (Figures represent a sample scenario with 3 training sub-datasets.)

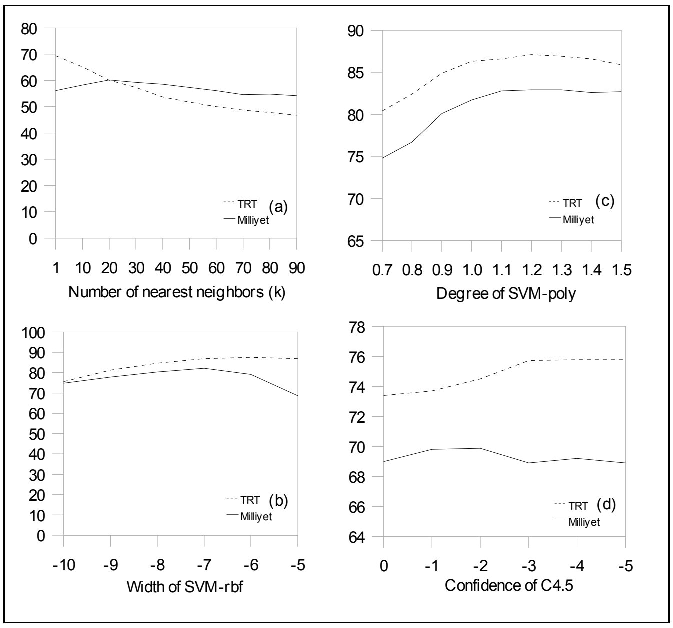

Figure 2. Parameter tuning results (*setup-1*) as accuracy vs. (a) k of KNN (b) Width of SVM-rbf. Default value is 0.01 (x axis value=$2^x$) (c) Degree of SVM-poly. Default is 1.0 (d) Confidence of C4.5. Default is 0.25 (x axis=$2^x$) (Figures are not drawn to the same scale.)

Performance of categorization is measured with accuracy in our study. Accuracy is given by the number of correctly classified documents divided by the number of all classified documents.

## IV. Experimental Results

### A. A Highly Accurate Setup

The experimental results given in this section are those of the optimized accuracies obtained after the final iteration. In the experiments, we observed at most three iterations. Firstly, parameter tuning results are given in Fig. 2. The value of number of k nearest neighbor is 20 and 1 using BilCat-MIL and BilCat-TRT respectively when the best accuracy values are obtained. The difference (20 vs. 1) is probably because of that BilCat-TRT is an imbalanced data set. SVM-rbf kernel performs the best when width is $2^{-7}$ and $2^{-6}$ on BilCat-MIL and BilCat-TRT respectively. SVM-poly kernel decides on 1.2 using both datasets. Lastly, confidence value of C4.5 are decided as default value ($2^{-2}$) and $2^{-4}$ on BilCat-MIL and BilCat-TRT respectively.

Term weighting results are given in Table III. Using KNN with *tf.idf* gives better results than other weighting approaches. The *tf* approach is not a good choice for NB and both SVM kernels. SVM-rbf works well with binary weighting. The results do not differ dramatically on C4.5.

Preprocessing results are given in Table IV. There is no word stemming and stopping applied in *none* setting. In the other settings, word stopping is applied with one of F*n* stemming. SVM-rbf and NB react positive to preprocessing on only BilCat-TRT. Preprocessing increases accuracies of other classifiers on both sets. The highest increase is seen in KNN.

TABLE III. TERM WEIGHTING RESULTS (*SETUP-2*) FOR ALL CLASSIFICATION APPROACHES ON BOTH DATASETS

| | *BilCat-MIL* | | | | | *BilCat-TRT* | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Weight Type: | KNN | SVM RBF | SVM Poly | C4.5 | NB | KNN | SVM RBF | SVM Poly | C4.5 | NB |
| Binary | 58.9 | **82.1** | 82.9 | 67.6 | 76.2 | 64,8 | **87.5** | **87.1** | 74.0 | 85.9 |
| TF | 56.1 | 69.3 | 79.8 | **69.8** | 67.2 | 65,8 | 74.6 | 84.3 | **75.7** | 81.9 |
| tf-idf | **60.2** | 77.5 | **83.1** | 69.7 | **77.4** | **69,4** | 84.1 | 86.6 | **75.7** | **86.9** |

TABLE IV. PREPROCESSING RESULTS (*SETUP-3*) FOR ALL CLASSIFICATION APPROACHES ON BOTH DATASETS.

| | *BilCat-MIL* | | | | | *BilCat-TRT* | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Stemmer | KNN | SVM RBF | SVM Poly | C4.5 | NB | KNN | SVM RBF | SVM Poly | C4.5 | NB |
| F3 | **60.2** | 82.0 | 83.1 | 67.1 | 67.5 | 66,9 | 85.5 | 85.4 | 72.7 | 81.3 |
| F4 | 57.9 | 80.9 | **83.3** | **69.8** | 71.7 | **69,4** | 86.6 | 86.2 | 72.9 | 85.2 |
| F5 | 56.5 | 81.1 | 83.1 | 68.7 | 73.0 | 67,4 | **87.5** | 86.5 | **75.7** | 86.6 |
| F6 | 52.0 | 80.8 | 81.5 | 67.8 | 73.5 | 64,2 | 87.1 | **87.1** | 74.1 | **86.9** |
| F7 | 50.2 | 81.5 | 81.2 | 64.3 | 76.2 | 65,4 | 86.3 | 87.0 | 74.0 | 86.2 |
| none | 50.8 | **82.1** | 80.5 | 65.0 | **77.4** | 61,8 | 84.7 | 83.9 | 70.8 | 84.4 |

TABLE V. SUMMARY OF ITERATIVE OPTIMIZATION FOR ALL CLASSIFICATION APPROACHES ON BOTH DATASETS

| | *BilCat-MIL* | | | | | *BilCat-TRT* | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | KNN | SVM rbf | SVM poly | C4.5 | NB | KNN | SVM rbf | SVM poly | C4.5 | NB |
| Initial accuracy | 24.8 | **82.1** | 80.3 | 66.0 | 73.7 | 38.9 | 83.1 | 83.5 | 67.2 | 82.5 |
| Optimized accuracy | **60.2** | **82.1** | **83.3** | **69.8** | **77.4** | **69.4** | **87.5** | **87.1** | **75.7** | **86.9** |

Feature selection results are given in Fig. 3. Selecting small number of features performs well with KNN because of the fact that nearest neighbor algorithms does not work well with high dimensions, which is called the curse of dimensionality [22]. On the other hand, selecting most of the features performs well with other classifiers. This is because of the fact that there are only few irrelevant features not to use in text categorization [12]. Information Gain and Chi-Squared performs better than others for smaller number of features using all classifiers. They can be used to increase efficiency without losing reasonable effectiveness.

Finally, summary of iterative optimization on both data sets is given in Table V. Initial accuracy obtained with default parameters and final optimized accuracy obtained at the end of the last iteration are listed for each classification methods. Default values are changed after deciding on a highly accurate setup on both data sets with all classifiers except SVM-rbf on BilCat-MIL. KNN is the most sensitive classifier to parameter changes. Its accuracy changes from 24.8 to 60.2 which is a 243% increase. Highest accuracies we achieve are 83.3 with SVM-poly and 87.5 with SVM-rbf on BilCat-MIL and BilCat-TRT respectively. Classifiers are more successful on BilCat-TRT in general. Naive Bayes performs approximately the same as SVM classifiers on BilCat-TRT. This can be explained by looking individual category accuracies. Naive Bayes performs better than SVM classifiers on the categories "Culture&Art" and "Health", which include smaller number of documents than other categories as Table II shows.

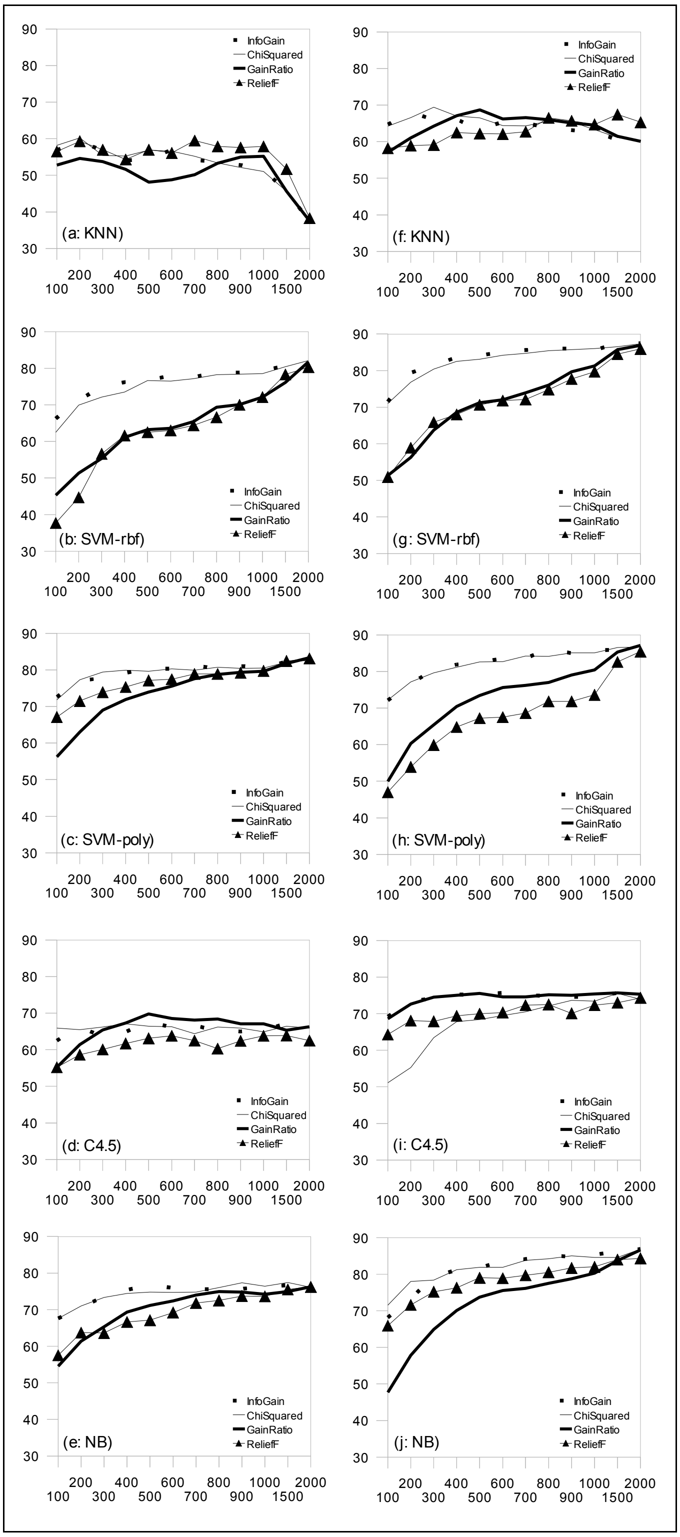

Figure 3. Parameter tuning results (setup-4) as accuracy vs. different number of features selected. (a) - (e) are conducted on BilCat-MIL. (f) – (j) on TRT.

## B. *Issues on News Portals*

*Changes in training data set size.* Results for the effect of changing train size are given in Fig. 4. Increasing the training size on both sets provides improvement on accuracy of C4.5

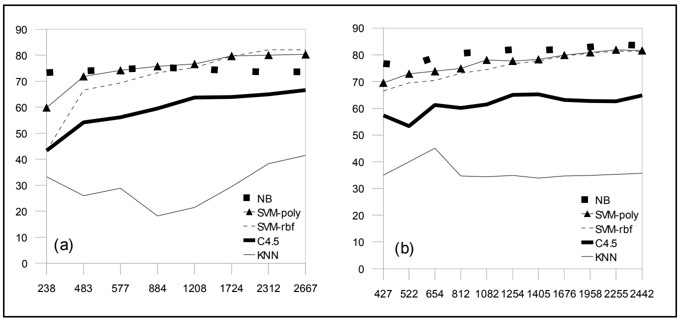

Figure 4. Effect of training size with different number of training sizes for all classifiers on two data sets. (a) BilCat-MIL (b) BilCat-TRT

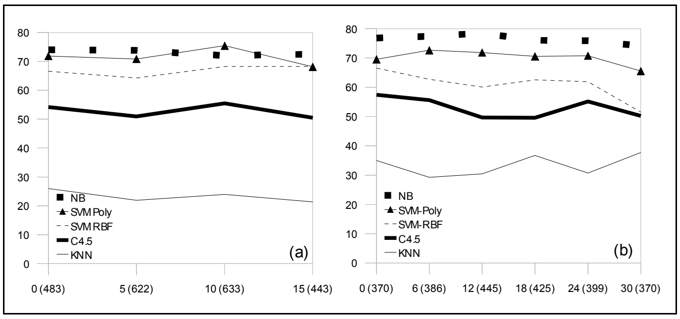

Figure 5. Robustness of classifiers by increasing min days between train and test sets (number of train documents) on two data sets. (a) BilCat-MIL (b) BilCat-TRT

and SVM with both kernels. However, KNN does not have a continuous accuracy increase. This can be due to the local character of KNN [3]. NB works well with small training sets. We explain it by its independence assumption that indicates each feature is independent of others. Therefore, it can easily make good estimates of probability in small sets [23].

*Changes in Classifier Robustness*. Robustness results are given in Fig. 5. The structures of datasets allow us to examine the effects of time distance between train and test sets at most 15 and 30 days in BilCat-MIL and BilCat-TRT, respectively. In Fig. 5-a, the x-axis value 15(443) means that time distance between train and test sets is 15 days including 443 documents. Considering our results on both data sets and assuming that small accuracy variations are unimportant, we can conclude that NB and SVM-poly are robust for approximately 30 and 10 days respectively. KNN, C4.5 and SVM-rbf are robust for a few days. NB is more robust than other classifiers probably due to its independence assumption explained before.

## V. Conclusion

In this paper we introduce a text categorization template for Turkish news articles. Our work develops a highly accurate categorization setup and examines issues related to text categorization on news portals. Our iterative optimization experiments may result in a local-maxima in parameter space. Testing all parameter combinations solves this problem; however it is inefficient. It would be interesting to examine some other term weighting schemes (e.g tf-rf, tf-icf). For the sake of efficiency, we employ F*n* stemmer in our study. However, some other stemming algorithms can also be examined.

## References

[1] Hayes P. J., Knecht L. E., Cellio M. J. (1988). A news story categorization system. *ANLP'1988*. 9-17.

[2] Yang Y., Liu, X. (1999). A re-examination of text categorization methods. *Proceedings of the 22nd Annual international ACM SIGIR Conference on Research and Development in information Retrieval.* 99:42-49.

[3] Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Comput. Surv. 34, 1:*1-47.

[4] Güran A., Akyokuş S., Güler N., Gürbüz Z. (2009). Turkish text categorization using n-gram words, *INISTA 2009*.

[5] Amasyalı M.F., Diri B. (2006). Automatic Turkish text categorization in terms of author, genre and gender. *In Proceedings of NLDB'2006.* 221-226.

[6] Amasyalı M.F., Yıldırım T. (2004). Otomatik haber metinleri sınıflandırma. *SIU'2004*. 224-226.

[7] Cover T., Hart P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27.

[8] Vapnik V. (1982). *Estimation of Dependences Based on Empirical Data* Springer-Verlag New York, Inc., Secaucus, NJ, USA.

[9] Platt J. (1998). Fast training of Support Vector Machines using Sequential Minimal Optimization. Advances in Kernel Methods - Support Vector Learning. *Eleventh Conference on Uncertainty in Artificial Intelligence*. pp. 338-345.

[10] Quinlan R. (1993). C4.5: Programs for Machine Learning. *Morgan Kaufmann Publishers*, San Mateo, CA.

[11] John G. H., Langley P. (1995). Estimating continuous distributions in Bayesian classifiers. *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*. pp. 338-345.

[12] Joachims T. (1998), Text categorization with support vector machines: learning with many relevant features. *Proceedings of the European Conference on Machine Learning.* pp. 137-142.

[13] Salton G., Buckley C. (1988). Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage*. 24(5): 513-523.

[14] Lovins J. B. (1968). Development of a stemming algorithm. *MTCL* 11:22–31.

[15] Can F., Kocberber S., Balcik E., Kaynak C., Ocalan H. C., Vursavas O. M. (2008). Information retrieval on Turkish texts. *Journal of the American Society for Information Science and Technology*. 59:407-421.

[16] Kononenko I. (1994). Estimating attributes: Analysis and extensions of RELIEF. *ECML'1994*. 171-182.

[17] Mitchell T. M. (1997). *Machine Learning*. The Mc-Graw-Hill Companies, Inc.

[18] Yang Y., Pedersen J.P. (1997). A comparative study on feature selection in text categorization. *ICML'1997* .412-420.

[19] Yang Y. (1996). Sampling strategies and learning efficiency in text categorization. *AAAI Spring Symposium on Machine Learning in Information Access 1996*. 88-95.

[20] Witten I. H., Frank E., Hall M. A. (2011). *Data Mining: Practical Machine Learning Tools and Techniques.* Morgan Kaufmann.

[21] Nash, J.F. (1950). Equilibrium points in n-person games. *In the Proceedings of the National Academy of the USA*. vol.36, pp. 48-49.

[22] Bellman R. (1961). *Adaptive Control Processes: A Guided Tour.* Princeton University Press, New York.

[23] Tan P. N., Steinbach M., Kumar V. (2005). *Introduction to Data Mining*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA.