Developing a Concept Extraction System for Turkish

Meryem Uzun-Per[1], **Hidayet Takçı** [2], **and Tunga Güngör**[1]
[1] Computer Engineering Department, Boğaziçi University, Bebek, Istanbul, Turkey
[2]Computer Engineering Department, GYTE, Gebze, Kocaeli, Turkey

Abstract - *In recent years, due to the vast amount of available electronic media and data, the necessity of analyzing electronic documents automatically was increased. In order to assess if a document contains valuable information or not, concepts, key phrases or main idea of the document have to be known. There are some studies on extracting key phrases or main ideas of documents for Turkish. However, to the best of our knowledge, there is no concept extraction system for Turkish although such systems exist for well-known languages. In this paper, a concept extraction system is proposed for Turkish. By applying some statistical and Natural Language Processing methods, documents are identified by concepts. As a result, the system generates concepts with 51% success, but it generates more concepts than it should be. Since concepts are abstract entities, in other words they do not have to be written in the texts as they appear, assigning concepts is a very difficult issue. Moreover, if we take into account the complexity of the Turkish language this result can be seen as quite satisfactory.*

Keywords: Concept Extraction, Natural Language Processing

1 Introduction

There is a rapidly growing amount of available electronic information such as online newspapers, journals, conference proceedings, Web sites, e-mails, etc. Using all these electronic information, controlling, indexing or searching is not feasible and possible for a human. For search engines, users have to know the keywords of the subject that they search, since search engines use a top down approach in order to find information in textual materials. The necessity of analyzing unstructured texts automatically is apparent. Users do not have to know the query terms and the main idea of the searched documents. If the concept of a document is known, a general knowledge about it also is known.

Concept is a term related to philosophy more than linguistics. In philosophy, a concept is defined as a thing apprehended by human thought and concepts are elements of thoughts and facts [1]. Concepts are different from words. Words are used for naming the concepts. It is possible that a single word can correspond to more than one concept or several words can define a single concept.

Concept extraction study aims at obtaining efficient solutions to some problems that are harder to solve using data mining. Crangle et al. define concept extraction as follows [2]:

> "Concept extraction is the process of deriving terms from natural-language text that are considered representative of what the text is about. The terms are natural-language words and phrases which may or may not themselves appear in the original text."

For concept extraction from unstructured texts there are mainly two approaches: expert-based and statistical. Expert-based approach has several disadvantages such as finding specialists on the subjects and developing learning based systems. In statistical approach, statistical methods are applied to the training data and models are built. Bayesian networks, neural networks, support vector machines, and latent semantic analysis are some of the statistical methods used in this area. Natural Language Processing (NLP) is different than these approaches in the sense that it uses the speed and cost effectiveness of the statistical approach but sometimes may require human intervention [3]. For linguistics-based approaches human intervention may be needed at the beginning to develop dictionaries for a particular industry or field of study. However, it has several considerable advantages such as getting more precise results quickly. Concepts can be extracted by using these models.

For English there are some studies done for concept extraction such as the studies of Crangle et al. [2] and Gelfand et al. [4], and there are some commercial software such as PASW Text Analytics and WordStat. These software also support several other languages such as French, Italian and Spanish. Moreover, there are some studies for unstructured Turkish documents for key phrase extraction such as [5] and [6]. However, key phrase extraction is different from concept extraction in the sense that key phrases are written in the documents as they appear, but concepts do not have to appear in the documents. There is neither a study on concept extraction nor software for Turkish. In this paper, a concept extraction system for Turkish is proposed.

2 Related Work

Concepts can be formed of words or phrases. Initially, sentences are divided into their words and phrases. For this

purpose, grammatical and syntactic methods are used which are tested in ontology learning, lexical extraction, and information retrieval systems [7]. In grammatical methods, if shallow parsing is used to parse the sentences, the whole sentence is converted into a grammatical tree where the leaves are noun and verb phrases. Then, noun phrases are selected as concepts. In syntactic methods punctuation and conjunctions are used as divisors. Then, all phrases are regarded as concepts. This approach is also used in keyword extraction systems [8].

For concept extraction there are two important application areas which are indexing documents and categorizing documents. Moreover, it is used for evaluating open ended survey questions [9], mapping student portfolios [7], extracting synonymy from biomedical data [2], extracting legal cases of juridical events [10], and several other areas. The main reason of the use of concept extraction in numerous fields is that concepts give an opportunity to enhance information retrieval systems [11, 12].

Extracting key phrases of documents is related to extracting concepts of documents. In academic articles, generally, key phrases are listed after the abstract which help the reader to understand the context of the documents before reading the whole document. Keyphrase Extraction Algorithm (KEA) is an automatic keyhrase extraction method that is proposed by Witten et al.[8]. The KEA was applied to Turkish documents by Pala and Cicekli by changing the stemmer and stop-words modules, and by adding a new feature to the algorithm [5]. Both for English and Turkish the success rates are about 25-30%.

In automatic key phrase extraction field a study is performed by Wang et al. [13] which uses neural networks for extracting key phrases. Turney uses two algorithms to extract key phrases from documents [14]. One of them is the C4.5 algorithm and the other is the GenEx algorithm. The overall success rate is very low. Rohini presented a study that extracts key phrases from electronic books by using language modeling approaches [15]. Kalaycılar and Cicekli [6] proposed an algorithm called TurKeyX for Turkish in order to extract key phrases of Turkish documents automatically which is based on statistical evaluation of noun phrases in a document. A study about extracting concepts automatically from plain texts is done by Gelfand et al. [4] by creating a directed graph called semantic relationship graph from WordNet. The success rate of all these studies is at most 30%.

There is some commercial software which is related to concept extraction. The two most popular software are PASW Text Analytics [3] and WordStat [16]. In Text Analytics linguistic resources are arranged in a hierarchy. At the highest level there are libraries, compiled resources and some advanced resources. Moreover, for English, there are specialized templates for some specific application areas like gene ontology, market intelligence, genomics, IT, and security intelligence. There are two types of dictionaries in libraries: compiled dictionaries which end users cannot modify and other dictionaries (type, exclusion, synonym, keyword, and global dictionaries) which end users can modify. The compiled dictionaries consist of lists of base forms with part-of-speech (POS) and lists of proper names like organizations, people, locations and product names. After extracting candidate terms, named entities and the dictionaries are used to identify concepts of documents. WordStat also uses the same principal while extracting concepts of the texts.

3 Concept Extraction System

3.1 Pre-processing

In order to develop a Concept Extraction System (CES) for Turkish, a corpus has to be determined to work on. The first step in this work is finding comprehensive Turkish documents. Then the pre-processing processes start. In order to run the codes on documents, all the documents have to be converted to text format. The text files are saved in UTF-8 format. Then, all documents in the corpus are tokenized such that a blank character is inserted before the punctuation characters.

3.2 Creating nouns list

Concepts can be determined from the nouns and the noun phrases. Therefore, in order to obtain the concepts of the documents, nouns in the documents have to be extracted. Extracting nouns of the documents and eliminating inflectional morphemes are difficult issues for Turkish. In this process, The Boun Morphological Parser (BoMorP) and The Boun Morphological Disambiguator (BoDis) programs [17] are used. They parse documents with an accuracy of 97%. They are applied to all the documents in the corpus. BoMorP parses the words and identifies their roots and morphemes. Turkish words are highly ambiguous in the sense that a single Turkish word can have several distinct parses. BoDis calculates a score for each parse according to the context. The output shows the root, the POS tag in square brackets, inflectional morphemes with '+' sign, derivational morphemes with '–' sign, and the score. The parse of an example word is as follows:

tekniklerin (*of the techniques*)
teknik[Noun]+lAr[A3pl]+[Pnon]+NHn[Gen] : 21.4033203125
teknik[Adj]-[Noun]+lAr[A3pl]+Hn[P2sg]+[Nom] : 19.7978515625
teknik[Adj]-[Noun]+lAr[A3pl]+[Pnon]+NHn[Gen]: 14.263671875
teknik[Noun]+lAr[A3pl]+Hn[P2sg]+[Nom] : 12.658203125

After the disambiguation process, the nouns in the documents are selected. If the parse with the highest probability is noun, it is selected unless it is an acronym,

abbreviation, or proper name. These types are also represented as noun in the root square bracket, but in the next square bracket their original type is written. So, the second square bracket is also checked in order to obtain the correct nouns list.

Inflectional morphemes are removed from the nouns. For example, the root forms of all the words "sistem, sistemler, sistemlerin, sistemde, sistemin, sisteme, etc." (*system, systems, of the systems, in the system, of the system, to the system,* etc.) are regarded as "sistem" and their frequencies are added to the "sistem" noun. However, derivational morphemes are kept as they appear. For example, the noun "delik" (*hole*) is derived from the verb "delmek" (*to drill*), however the noun "delik" is added to the nouns list in this form. All nouns are listed for the documents and their frequencies are calculated. Then all nouns are stored in one file, the same words in the documents are merged, and their frequencies are added. The nouns that occur in documents rarely are considered as they cannot give the main idea of them. If the frequencies of the nouns are less than three, they are eliminated to decrease the size of the list and speed up later processing.

3.3 Clustering cumulative nouns list

Concepts can be defined by nouns. Therefore, clustering similar nouns is helpful in order to determine the concepts. For this purpose, some clustering methods such as hierarchical clustering and k-means clustering are applied to the cumulative nouns list. These clustering methods are unsupervised learning algorithms which do not need any training step to pre-define the categories and label the documents.

First of all, document-noun matrix is created from the cumulative nouns list, which holds the documents in rows and the nouns in columns, and the intersection of a row and a column gives the number of times that noun appears in the document. The clustering algorithms are applied to the matrix for different numbers of clusters such as 10, 25, 50, 75 and 100. Hierarchical clustering algorithms are coded by MATLAB, k-means algorithm is applied by Tanagra [18]. Clusters are assessed by human specialists. It is observed that the k-means algorithm for 100 clusters performs much better than the other possibilities.

3.4 Assigning clusters to documents

After the clustering operation, the clusters are assigned to the documents. This is done by searching the nouns of the documents in the words of the clusters. A ratio is calculated for each possible cluster of a document by dividing the number of the words in the possible cluster of the document to the number of the words in that cluster. If the ratio is more than a threshold value, the cluster is assigned to the document. So, it can be said that this document can be defined by that cluster. The threshold is selected as "1"; in other words, if a document contains all the words of a cluster, this cluster is assigned to that document. Because, it is observed that if a document is related to a cluster it should contain all the words of that cluster. More than one cluster can be assigned to a document. Similarly, a cluster can be assigned to more than one document. Figure 1 shows the pseudo-code of assigning clusters to documents.

```
Input
    F1: Documents-Words file
    F2: Clusters-Words file
Output
    F3: Documents-Clusters file
Begin
1:  L1 <- Read F1 to list
2:  L2 <- Read F2 to list
3:  for each word w in L1
4:      Search cluster cl of w in L2
5:      Append cl to L1
6:  end for
7:  for each document d
8:      L3 <- Read clusters of d in L1
9:      L4 <- Read words of d in L1
10:     for each cluster cl in L3
11:         A <- Number of words in cl in L4
12:         B <- Number of words in cl
13:         if (A/B >= Threshold)
14:             Write d + cl to F3
15:         end if
16:     end for
17: end for
End
```

Figure 1: Assigning clusters to documents

3.5 Identifying documents by concepts

The main aim of this study is to define documents with concepts. Therefore, a transition has to be done from words and clusters to concepts. In concept extraction software like PASW Text Analytics and WordStat, dictionaries are used in order to identify documents by concepts as mentioned before [3, 16]. These dictionaries consist of concepts and words related to these concepts. In both programs, users can add or remove concept categories or words to the categories. Similar to these programs, it is decided to create concept categories and words related to them. So, concepts have to be assigned to clusters according to the words they contain by human specialists. Then, concepts are assigned to the documents according to their assigned clusters. Figure 2 shows the module for assigning concepts to documents.

```
Input
    F1: Documents-Clusters file
    F2: Clusters-Concepts file
Output
    F3: Documents-Concepts-Count file
Begin
1:  L1 <- Read F1 to list
2:  L2 <- Read F2 to list
3:  for each document i
4:      L3 <- Read clusters of i
5:      L4 <- empty
6:      for each cluster cl in L3
7:          L5 <- read concepts of cl
8:          for each concept c in L5
9:              if (L4 does not contain c)
10:                 Add c + "1" to L4
11:             else
12:                 Increase count of c in L4
13:             end if
14:         end for
15:     end for
16:     Write L4 to F3
17: end for
End
```

Figure 2: Assigning concepts to documents

4 Experiments and Evaluations

4.1 Corpus Selection

In order to develop a CES for Turkish, a corpus is needed to work on. The first step in this work is finding comprehensive Turkish documents. Online archives of the Journal of the Faculty of Engineering and Architecture of Gazi University [19] are selected as a corpus which is also used in [5] and [6]. It contains 60 Turkish articles and 60 key files which contain the keywords of the articles.

4.2 Testing Methodology

After selecting a corpus the methodology is applied to the corpus by following the steps explained before. Then several tests are applied to the results obtained. These tests are test by words, test by clusters, and test by concepts. Precision and recall are used in order to measure the success rates which are widely used metrics to evaluate the correctness of results of data mining projects. Equations 1 and 2 show the formula of precision and recall, respectively, where *a* is the number of retrieved and relevant records, *b* is the number of retrieved records, and *c* is the number of relevant records [20].

$$precision = \frac{a}{a+b} \quad (1)$$

$$recall = \frac{a}{a+c} \quad (2)$$

4.3 Test by words

Correctness of the clusters which are assigned to the articles is tested by words via the words of the key files. If the clusters are created and assigned correctly, the words in the clusters which are assigned to the articles should match with the nouns in the key files. We denote the words of the clusters which are assigned to an article as *w1* and the nouns in the key file of that article as *w2*. *w2* is searched in *w1*. For each article, the numbers *w1*, *w2*, and the intersection of *w1* and *w2* are calculated. Here precision is not needed to be calculated because clusters contain a lot of words and limiting them is not possible in this methodology. Only recall is calculated. According to Equation 2; *a* is the number of the intersection of *w1* and *w2*, *(a + c)* is the number of *w2*.

Average recall is calculated as 0.46. About half of the nouns of the key files are contained in the words of the clusters which are assigned to the articles. This information cannot explain the accuracy of the study because the clusters contain a lot of words in them; however the words of the key files are very limited. But unfortunately, although a lot of nouns are selected from the articles, only half of them are matched with the nouns of the key files. Figure 3 shows the number of the nouns of the key files versus the number of the matched nouns for the documents.

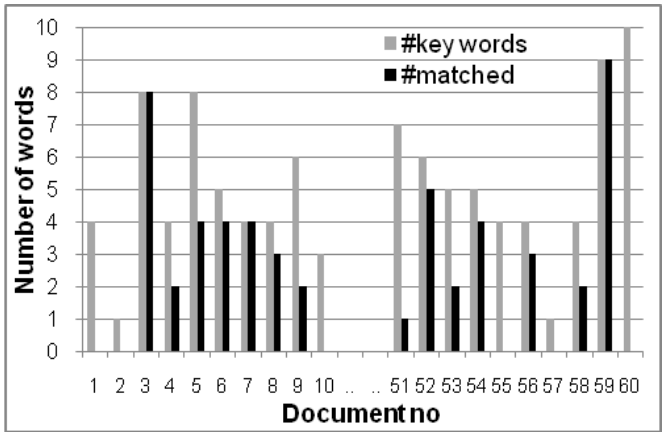

Figure 3: Number of key words versus number of matched words

4.4 Test by Clusters

Correctness of the clusters which are assigned to the articles is tested by clusters via the clusters of the key files. Clusters are assigned to the key files according to the nouns in them. We denote the clusters of an article as *cl1* and the clusters of the key file related to that article as *cl2*. *cl1* and *cl2* are compared. For each article, the numbers *cl1*, *cl2*, and the intersection of *cl1* and *cl2* are calculated. Then, precision and recall are calculated for each document. According to Equations 1 and 2; *a* is the number of the intersection of *cl1* and *cl2*, *(a + b)* is the number of *cl1*, and *(a + c)* is the number of *cl2*.

Average precision and average recall are calculated as 0.50 and 0.41, respectively. As a result of the test by clusters, 41% of the assigned clusters are matched with the clusters of the key files. Half of the clusters which are assigned to the articles are assigned correctly. The recall is lower than expected. Since the clusters are considered as general topics of the articles, it indicates that the general topics of the articles cannot be determined perfectly. However, for Turkish it can be regarded as a success because of the complexity of the language. Figure 4 shows the number of the clusters of the key files versus the number of the matched clusters for the articles.

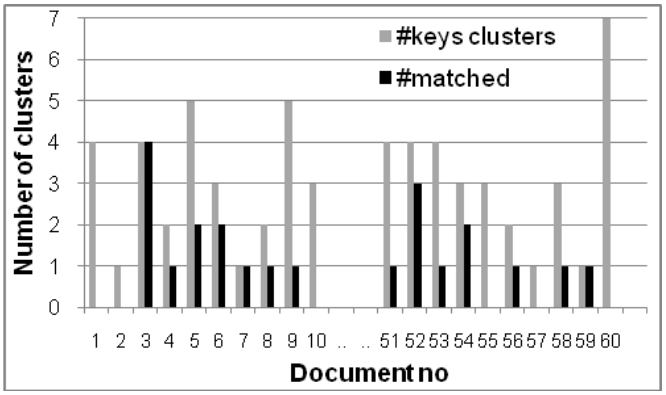

Figure 4: Number of clusters of the key files versus number of the matched clusters

4.5 Test by Concepts

Correctness of the concepts which are assigned to the articles is tested by concepts via the concepts of the key files. We denote the concepts which are assigned to an article as *c1* and the concepts of the key file related to that article as *c2*. *c1* and *c2* are compared. For each article, the numbers *c1*, *c2*, and the intersection of *c1* and *c2* are calculated. Then, precision and recall are calculated for each article. According to Equations 1 and 2; *a* is the number of the intersection of *c1* and *c2*, *(a + b)* is the number of *c1*, and *(a + c)* is the number of *c2*.

Average precision and average recall are calculated as 0.22 and 0.51, respectively. As a result of the test by concepts, 51% of the concepts which are assigned to the articles are matched with the concepts of the key files. 22% of the concepts which are assigned to the articles are assigned correctly. This shows that more concepts are assigned than it should be. The recall being too high may be due to this fact. Since concepts are abstract entities, in other words they do not have to be written in the texts as they occur, assigning concepts is a very difficult issue. Furthermore, Turkish is an agglutinative and complex language that studies on Turkish do not give high scores. For example, the success rate of key phrase extraction studies [5] and [6] are not passed over 30%. As the first study for Turkish in this subject, the results can be seen as quite successful. Figure 5 shows the number of the concepts of the key files versus the number of the matched concepts for the articles.

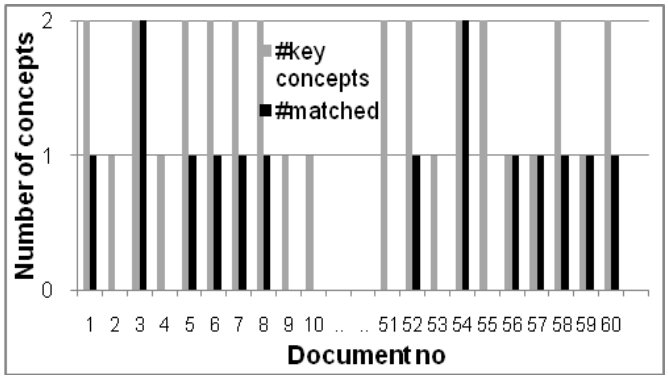

Figure 5: Number of key concepts versus number of the matched concepts

As a result of the test by concepts, precision is considered as low; therefore it is thought that limiting the number of the concepts assigned to the articles may be useful for the results. Due to the similarity of the clusters, some clusters contain the same concepts. So, while assigning concepts to the articles via clusters, some concepts are assigned to the articles more than once. Therefore, we performed another experiment in which a restriction is applied to the concepts of the articles such that if an article is defined by a concept more than once, the concepts that exist only once are eliminated. If an article is defined by concepts only once, no elimination is applied. For evaluation, the same formulas are applied which are explained in the test by concepts. Average precision and average recall are calculated as 0.16 and 0.27, respectively. Both precision and recall decrease significantly. By applying this test, precision is expected to be increased however it decreases. Moreover, recall decreases drastically. If we eliminate all the concepts of the articles which exist only once to define the articles, the results get worse. This shows that the results are much better without any elimination. Therefore, the result of this test can be given as 51% recall with 22% precision.

5 Conclusions

In this paper, a concept extraction system for Turkish is proposed. The first issue that must be faced is the complexity of Turkish which is an agglutinative language. The second issue is the abstractness of concepts. To the best of our knowledge, this study is the first concept extraction study for Turkish. This work can serve as a pioneering work in concept extraction field for agglutinative languages. The results are better than the studies related to this field.

As a future work, the methodology must be applied to new corpora in different domains. In order to improve the methodology, other clustering methods such as supervised learning algorithms can be tried.

6 Acknowledgements

This work is supported by the Boğaziçi University Research Fund under the grant number 5187, the Scientific and Technological Research Council of Turkey (TÜBİTAK) under the grant number 110E162. Meryem Uzun-Per is supported by TÜBİTAK BİDEB 2210. The authors would like to thank to İlyas Çiçekli for the data set.

7 References

[1] Mengüşoğlu, T. 1992. Felsefeye Giriş 7. Ed. Istanbul: Remzi Kitabevi.

[2] Crangle, C.; Zbyslaw, A.; Cherry, M.; and Hong, E. L. 2004. Concept Extraction and Synonymy Management for Biomedical Information Retrieval. In Proceedings of the Thirteenth Text REtrievel Conference. Gaithersburg, MD: National Institute of and Technology.

[3] SPSS Inc. 2009. Mastering new challenges in text analytics, Technical Report, MCTWP-0109.

[4] Gelfand, B.; Wulfekuhler, M.; and Punch W.F. III. 1998. Automated Concept Extraction from Plain Text, Technical Report, WS-98-05, AAAI.

[5] Pala, N. and Cicekli, I. 2007. Turkish Keyphrase Extraction Using KEA. In Proceedings of 22nd International Symposium on Computer and Information Sciences, 193-197, Ankara, Turkey.

[6] Kalaycılar, F. and Cicekli, I. 2008. TurKeyX: Turkish Keyphrase Extractor. In 23rd of the International Symposium on Computer and Information Sciences. Istanbul, Turkey.

[7] Villalon, J. and Calvo, R.A. 2009. Concept Extraction from Student Essays, Towards Concept Map Mining. In Proceedings of the 2009 Ninth IEEE International Conference on Advanced Learning Technologies, 221-225. IEEE Computer Society, Washington DC, USA.

[8] Witten, I.H.; Paynter, G.W.; Frank, E.; Gutwin, C.; and Nevill-Manning, C. G.. 2000. KEA: Practical Automatic Keyphrase Extraction. Computer Science Working Paper 00/05. Hamilton, New Zealand: University of Waikato.

[9] SPSS Inc. 2008. Gaining Full Value from SPSS Text Analysis for Surveys, Technical Report, GVSTWP-1008.

[10] Moens, M.F. and Angheluta, R. 2008. Concept Extraction from Legal Cases: The Use of a Statistic of Coincidence. In Proceedings of the Ninth International Conference of Artificial Intelligence and Law, 142-146. New York, NY, USA: ACM.

[11] Bing, J. 1987. Performance of Legal Text Retrieval Systems: The Curse of Boole. Law Library Journal 79:187-202.

[12] Rissland, E. L.; Skalak, D. B.; and Friedman, M.T. 1996. Bankxx: Supporting legal arguments through heuristic retrieval. Artificial Intelligence and Law 4 (1):1-71.

[13] Wang, J.; Peng, H.; and Hu, J. 2006. Automatic Keyphrase Extraction from Document Using Neural Network. Advances in Machine Learning and Cybernetics. Berlin, Heidelberg, Springer-Verlag LNAI 3930:633-641.

[14] Turney, P. 1999. Learning to Extract Keyphrases from Text, Technical Report, ERB-1057, National Research Council, Institute for Information Technology, Canada.

[15] Rohini, U. and Ambati, V. 2007. Extracting Keyphrases from books using language modeling approaches. In Proceedings of the 3rd ICUDL, Pittsburgh, USA.

[16] Provalis Research. 2009. WordStat v6.0 Content Analysis and Text Mining, Help File, Montreal, Canada.

[17] Sak, H.; Güngör, T.; and Saraçlar, M. 2008. Turkish Language Resources: Morphological Parser, Morphological Disambiguator and Web Corpus. In GoTAL 2008, LNCS 5221:417-427. Springer.

[18] Rakotomalala, R. 2005. TANAGRA: A Free Software for Research and Academic Purposes. In Proceedings of EGC 2005, FRNTI-E-3, 2:697-702. Lyon, France.

[19] 2006. Journal of The Faculty of Engineering and Architecture of Gazi University 20(1-3), 21(1-4).

[20] Alpaydın, E. 2010. Introduction to Machine Learning, 2e. London, England: The MIT Press.