Research Paper **Open Access**

Text Mining Assistant

Muslum Serdar Akis[1], Semih Utku[2*]
[1](Computer Engineering Department/ Dokuz Eylul University, Izmir, Turkey)
[2](Computer Engineering Department/ Dokuz Eylul University, Izmir, Turkey)
*Corresponding author: semih@cs.deu.edu.tr

ABSTRACT : *Text mining applications have become very popular in different areas and on the other hand, text mining is a time-consuming task. In each different study, the text mining steps are repeated more than once in the pre-process and modeling steps. There is no tool to assist the researcher in these stages. In our study, text mining methods were examined and repetitive steps were determined. It was thought that, these identified steps could be performed with the help of an assistant tool therefore time loss could be prevented. Within the scope of the study, an application has been developed to complete these repetitive steps in a short time and to use them repeatedly. This application which was developed with Microsoft .Net can work stably in many different systems; it is aimed that it can be used in different studies with improved settings. This application, which was developed especially for the Turkish language, aims to investigate the patterns in the texts and avoid repetitive steps in the studies.*

Keywords -*Text Mining, Data Mining, Data Analysis, Software*

I. INTRODUCTION

A lot of information has been recorded in our daily lives. After computers entered into our lives, it became much easier to edit and save these records. However, information alone is of no use for any aim. The information obtained should be processed for a purpose. Otherwise the recorded information is nothing but useless memory loss. Considering that today many information is stored on texts; the fact that the information is idle is revealed as a result of the evaluation of this data. With the help of text mining, the information stored in these texts can be revealed.

Text mining is a laborious and time consuming process. In particular, since the texts are not directly in a structured form, they must first be processed and separated from unnecessary information. This information, which does not directly affect the outcome, makes it difficult to draw conclusions. However, in text mining, the preprocessing phase is important and it is most likely to make mistakes. In addition, these processes have to be tried again and again every time a long time is spent. Given these reasons, it has become necessary to carry out these operations by the help of a program. It has been evaluated that these operations can be made easier, faster and repeatable via an auxiliary tool that is to be developed.

In the field of text mining, many different algorithms are analysed [1]. These algorithms try to identify meaningful and usable patterns in the text. The most basic stage of the Text Mining process is preprocessing. Preprocessing processes include tokenization, filtering, lemmatization and stemming. After these stages, text mining processes are carried out by different methods.

It is seen that there are studies in many fields related to Text Mining [2]. In general, in the field of health [3], in agriculture [4], in the field of social network [5], in the financial field [6] and it is used in many other fields. One of the most basic operations in this field is the development of a software tool for public use. PubTator in the field of health; [7] Wei et.al have developed a web-based text mining tool. an Internet-based hypertext program was developed with MedMiner assistant tool [8]. The developed application was used for Gene Expression Profiling over text data. In another study [9] have developed an analytical tool that can be used on patent data. It shows the overall relationship among patents as a visual network. Samir et.al developed a text mining tool in their work for detecting associations of micro RNAs with Diseases. [10] is an automated literature mining tools which is used to find microRNA-disease associations. When the studies in the literature are examined, it is seen that there are many special purpose or general purpose text mining assistant.

Within the scope of this study, a Text Mining Assistant (TMA) was developed which could help text mining professionals and at least keep them away from repetitive steps while they are concentrating on the details of the their studies. Random TMA, Sequential Minimal Optimization (SMO) and Iterative Classifier Optimizer algorithms were added to the developed TMA.

II. TEXT MINING

The first text mining studies started in 1980 [11]. Considering the limited technologies of the period, text mining had its golden age in the 2000s. Text mining is the process of automatically extracting information and patterns on texts that were not previously recognized by computers [12]. The general processes of the text mining is shown in Fig. 1.

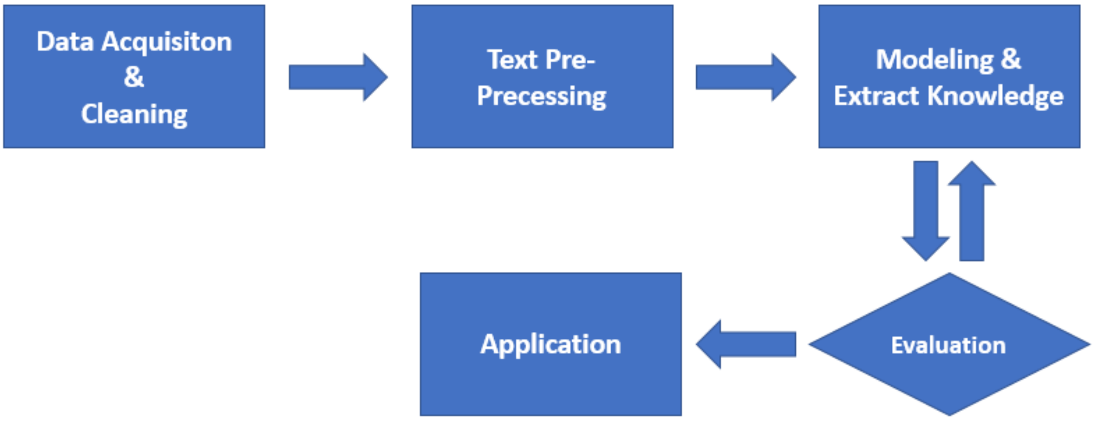


Fig. 1:Text Mining Processes

Text mining is actually not much different than data mining. The main difference is that data mining examines digitized data, while text mining examines non-digitized data, such as websites, html files and e-mails [13]. In text mining, data should be pre-processed with intermediate algorithms and text clearing methods before being examined.

Text mining operations must be performed repeatedly in each different data set and in each different method to be tested. In addition, these processes are long and laborious tasks that a lot of time for each and every operation. For this reason, it is necessary to develop an application at this stage and to perform this process in a parametric manner and to use it again and again.

The pre-processing stage of text mining is one of the most important stages of text mining. Errors to be made at this stage also affect other processes. In the pre-stage, meaningless characters are deleted from the text. Words that do not affect the result are removed from the text. Prefixes and suffixes, if any, are omitted. With the proposed study, it is aimed to prevent the errors that may occur by automating the processes at this stage and to minimize the loss of time.

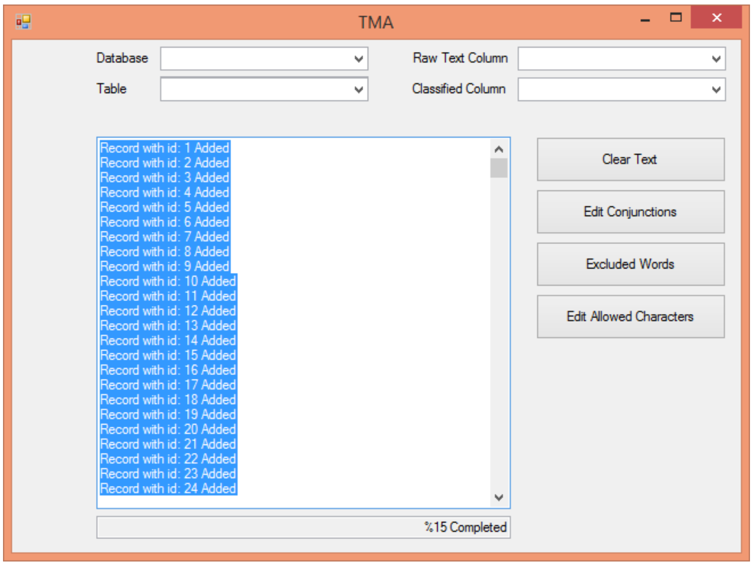
Once the texts have been cleared, the proposed tool also assists the user in modeling and evaluation phases. The application provides estimation by using data mining methods. Thus, repetitive tasks are automated and as a consequence, the one who conducts this study is given the chance to focus on the key aspects.

III. TEXT MINNING ASSISTANT

The Text Mining Assistant is a helpful application which is designed to help the person use text mining methods. It is aimed to eliminate routine strenuous jobs in text mining. These routines are eliminated, allowing the user to focus on more important parts. TMA has been specially developed for Turkish language. In addition, WEKA [14] has strengthened the unified application to enable researchers to easily access the latest technologies in machine learning.

The greatest waste of time during text mining tests and applications is due to the pre-processing stage. The resulting unstructured data should be cleared according to predetermined rules, but there is no specialized program for this process. The application was developed to eliminate the loss of time in text mining. The application parametrically processes the data and allows the cleaned information to be exported to the files ("arff" extension) supported by the user's WEKA.

When the application starts, the Login screen is displayed to connect the user to the database containing the data to be cleaned. In this field, after entering the necessary information, by pressing "Connect" button; database connection is established. There are two different login options: SQL Server Authentication and Windows Authentication. The application currently supports only MsSql database. After connecting to the database, the user is requested to input the table and column information containing the data to be cleaned. After the appropriate column is added, the data is cleared and added to the application's own database. This process prevents the same records from being cleaned repeatedly. In addition, the necessary classification information is recorded in the database. The application screenshot is shown in Fig. 2.



```
Record with id: 1 Added
Record with id: 2 Added
Record with id: 3 Added
Record with id: 4 Added
Record with id: 5 Added
Record with id: 6 Added
Record with id: 7 Added
Record with id: 8 Added
Record with id: 9 Added
Record with id: 10 Added
Record with id: 11 Added
Record with id: 12 Added
Record with id: 13 Added
Record with id: 14 Added
Record with id: 15 Added
Record with id: 16 Added
Record with id: 17 Added
Record with id: 18 Added
Record with id: 19 Added
Record with id: 20 Added
Record with id: 21 Added
Record with id: 22 Added
Record with id: 23 Added
Record with id: 24 Added
```

%15 Completed

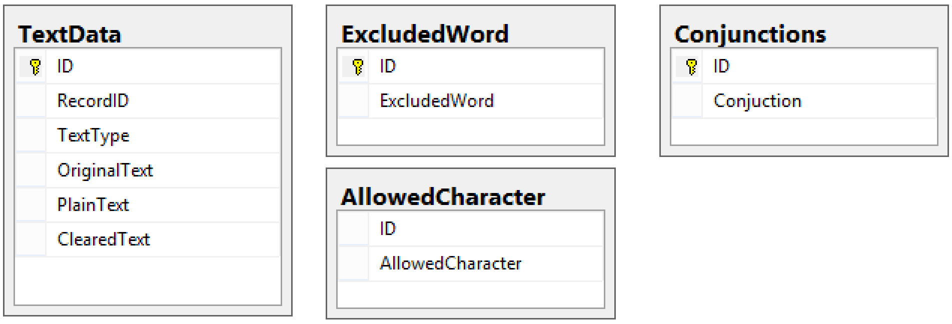
Fig. 2:Text Convert Screen

Users can use “Clear Text” button; The TMA cleans the texts according to the selected information and saves the cleaned states to the application database. When clearing the data, all data is scanned and if any characters are included in the external text, the allowed characters are deleted from the text. Allowed characters can be edited with the "Edit Allowed Characters" button.

After the text is cleared of special characters, the next step is to simplify the text by clearing the conjunctions and words that do not affect the result. Accordingly, the list of conjunctions obtained from the Turkish Language Association [15] is added by default. However, the user can add new words to this field and remove existing words if he/she wishes. Conjunctions to be checked can be customized from the page that opens with the “Edit Conjuctions” button. According to the information received from TDK, 35 conjunctions are added to the system by default.

Each word in the text is checked one by one and cleared from the text in case of any match. After the texts have been cleared of special characters and conjunctions, the next step of the process is to clear the vocabulary from any prefix or suffix. Zemberek Library [16] was used in this stage. Zemberek is a natural language processing library specially developed for Turkish. Supported by Java and C #, this library is a very convenient and popular tool for Turkish language processing. Separating words from suffixes and prefixes is a more complex task compared to other preprocessing tasks. First, the text, separated from special characters and links, is fragmented by the Zemberek library and finds the root of each word. The root of this word is used for further calculations. For words with more than one meaning, the first meaning in the dictionary is considered valid.

Simplification of texts is a very long process. Since there are huge amount of text in total, preprocess cannot be done often to examine and clear all texts with the above mentioned algorithms. Therefore, the texts are examined, and the information obtained is saved in the MSSQL database. This step prior to digitization is done to prevent any time loss that could occur in each trial. Database model is shown in Fig. 3.



TextData
- ID
- RecordID
- TextType
- OriginalText
- PlainText
- ClearedText

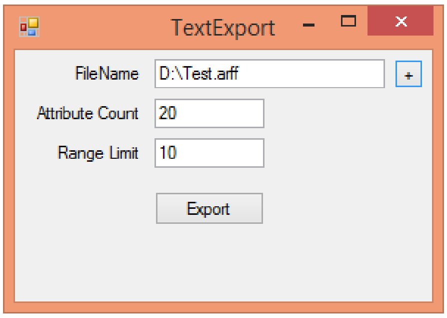
ExcludedWord
- ID
- ExcludedWord

AllowedCharacter
- ID
- AllowedCharacter

Conjunctions
- ID
- Conjuction

Fig. 3:Basic Database Design

Once the texts are simplified, they are ready to be digitized. In this context, the words in the texts are examined and the frequency of each word in the text is outputted. The frequency range is taken dynamically from user interface. First, all texts are scanned, and the frequency of each word is calculated. Obtained frequencies according to the desired attribute number is examined and the resulting matrix is obtained from the most frequent words. The resulting matrix is combined with the result field information to obtain ready-to-learn files. The export operation of the application is shown in Fig. 4.



TextExport

FileName: D:\Test.arff

Attribute Count: 20

Range Limit: 10

Fig. 4: Export Operation

In this section, there are parameters related to digitized data and Export button. After entering the file name to record the data, the number of attributes to select from each subject and frequency max limit are specified. The "Attribute Count" field represents the number of words to select from the text. If two is entered, the most frequently used two words are selected and the file is created accordingly. "Range Limit" corresponds to the frequency limit of the words in the text. For example, if two is entered, the frequency will be left as two if the frequency is two or more.

The text convert application was developed using Winform and .Net frameworks. Also powered by MSSQL data, this application has the capacity to analyze millions of patient records and generate the necessary results. In addition, each step was tested to ensure the accuracy of the digitized data generated. After the texts are cleared and ready for the learning process, another step is to implement data mining by using gathered data.

IV. TMA DATA MINING MODULE

After the creation of texts and arff files, the application is ready for the mining process. Experiments can be made with text mining algorithms by using these files which are exported in the application. TMA allows to use more than one algorithm together that can be set parametrically in the application. The application supports popular algorithms in the current case. The flexible and extensible existing structure and weka-powered TMA can also be made to support many algorithms in a short time. These three popular algorithms are listed below.

- Random Committee
- Sequential Minimal Optimization (SMO)
- Iterative Classifier Optimizer

In the random committee classifier, the seeds which is generating different random numbers build various base classifiers. The individual base classifier determines the final prediction using a straight average of the predictions for each base classifier [17] Sequential Minimal Optimization (SMO) is a simple algorithm that can quickly solve the SVM QP (Quatratic Programming) problem without any extra matrix storage and without using numerical QP optimization steps at all [18]. In iterative classification, a model is built using a variety of static and dynamic attributes. Classifier that includes dynamic attributes rely on the previous classification of related objects. When training the model, the class labels of all objects are known and consequently the values of all dynamic attributes are also known [19].

V. CONCLUSION

As a result of this study, the text cleaning process, which may take a long time, has become much shorter. The pre-processing steps are performed automatically, therefore there is less chance of encountering any problems and errors at this stage. Since the application was developed using .Net and C#, it was made portable to work on various hardware.

The developed application is able to work with very large data. It is capable of examining and clearing millions of records. The developed application is a tool that can be of great help especially in the early stages of Text mining. Only the extension .arff is supported at the moment, however it can be improved further in prospective versions. The data mining experiments after the text clearing process can be used to examine the accuracy of the methods used. In this way, time consumed in these stages can be reduced to shorter periods. The time and energy used may eliminate the routines performed before each operation, leaving more time for the actual work required.

REFERENCES

[1] M.Allahyari, S. Pouriyeh, M. Assefi,S. Safaei, E. D. Trippe,J. B. Gutierrez, and K. Kochut, A brief survey of text mining: Classification, clustering and extraction techniques. *arXiv preprint arXiv:1707.02919*, 2017.

[2] R. Alghamdi, and K. Alfalqi, A survey of topic modeling in text mining. *Int. J. Adv. Comput. Sci. Appl.(IJACSA), 6(1)*, 2015.

[3] R. Feldman, O. Netzer, A. Peretzand B. Rosenfeld, Utilizing text mining on online medical forums to predict label change due to adverse drug reactions, *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining, 2015,* ACM 1779-1788.

[4] S. A. Salloum, M. Al-Emran, A.A. Monem and K. Shaalan, A survey of text mining in social media: facebook and twitter perspectives. *Adv. Sci. Technol. Eng. Syst. J, 2(1)*,2017, 127-133.

[5] R. Irfan, C. K. King, D. Grages, S. Ewen, A survey on text mining in social networks, *The Knowledge Engineering Review*, *30(*2),2015, 157-170.

[6] B. S. Kumarand V. Ravi, A survey of the applications of text mining in financial domain, *Knowledge-Based Systems*, *114*, 2016, 128-147.

[7] C. H. Wei, H. Y. Kao and Z. Lu, PubTator: a web-based text mining tool for assisting biocuration. *Nucleic acids research*, *41*,W1,2013, 518-522.

[8] L. Tanabe, U. Scherf,L. H. Smith, J. K. Lee, L. Hunter and J.N. Weinstein, MedMiner: an Internet text-mining tool for biomedical information, with application to gene expression profiling. *Biotechniques*, *27*(6),1999, 1210-1217.

[9] B. Yoon and Y. Park, A text-mining-based patent network: Analytical tool for high-technology trend. *The Journal of High Technology Management Research*, *15*(1),2004, 37-50.

[10] S. Gupta, K.E Ross, C.O Tudor, C. H. Wu, C. J. Schmidtand K. Vijay-Shanker, miRiaD: A text mining tool for detecting associations of micrornas with diseases. *Journal of biomedical semantics*, *7*(1),2016, 9.

[11] P. Schulman,C. Castellonand M.E. Seligman, Assessing explanatory style: The content analysis of verbatim explanations and the Attributional Style Questionnaire. *Behaviour research and therapy*, *27*(5), 1989,505-509.

[12] M. A. Hearst, Untangling text data mining. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics.* 3(10), 1999.

[13] V. Gupta and G. S. Lehal, A survey of text mining techniques and applications. *Journal of emerging technologies in web intelligence*, *1(1)*, 2009, 60-76.

[14] M. Hall,E. Frank, G. Holmes, B. Pfahringer,P. Reutemannand I.H. Witten, The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter*, *11(*1), 2009, 10-18.

[15] TDK, Türk Dil Kurumu, *TDK 2019.*

[16] A. A. Akın, and M. D. Akın, Zemberek, an open source nlp framework for turkic languages. *Structure*, *10*, 2007, 1-5.

[17] Y. M. Huang,C. M. Hung and H. C. Jiau, Evaluation of neural networks and data mining methods on a credit assessment task for class imbalance problem. *Nonlinear Analysis: Real World Applications*, *7*(4),2006, 720-747.

[18] J. Platt, Sequential minimal optimization: A fast algorithm for training support vector machines. *Advances in kernel methods*,1998, 185–208.

[19] J. Neville, Iterative Classification: Applying Bayesian Classifiers in Relational Data. *Proc. AAAI-2000 Workshop on Learning Statistical Models from Relational Data*,2000,42–49