Malay Document Analysis and Recognition

NORZAIDAH MD NOH[1], MOHD RUSYDI ABDUL TALIB[2], AZLIN AHMAD[1],
SHAMIMI A. HALIM[1], AZLINAH MOHAMED[3]

[1] Lecturer, SIG Intelligent Systems; [2] BSc (IS) Graduate; [3]Associate Professor, Ph.D
Faculty of Information Technology and Quantitative Science
Universiti Teknologi MARA, 40800 Shah Alam, Selangor, MALAYSIA
Tel/SMS : +6016 3388840 Fax : +603-41072262 e-mail : (norzaidah, azlin, shamimi, azlinah}@tmsk.uitm.edu.my

Abstract:- Malay Document Analysis and Recongition aims to extract digital malay documents automaticaly. These extracted documents are presented in the form of namely articles, newspapers and magazines. Over the years, Malay digital documents has increased and published on the world-wide-web (www) and consequently used by many organizations local and abroad. In this paper, we introduce the implementation of a tool for Malay language document identification in mono- and multi-lingual documents. The tool development includes a feature extraction and a neural network technique. The feature extraction consists of documents filtering, word matching and binary representation of varied length sentences from many types of documents including generic text files, MS Word files, Adobe PDF and HTML web pages. The neural network employs back propagation neural network (BPNN) algorithm with adjustable number of neurons and weights between input, hidden and output layer. A database was constructed consisting of 300 sentences of mono and multi-lingual documents. Experiments show average recognition rate of 90% accuracy in recognizing of Malay language documents, which has more than 80%, matched Malay words. Our tool is able to recognise Malay language documents with reasonable accuracy.

Keywords: - Document processing, Language recognition, Backpropagation neural network, Document filtering, Word matching technique

1 Introduction

The widespread use of computers and the rise of world-wide-web (www) enable digital documents to be disseminated to virtually endless number of sites on the global network. Not to mention the continuous and rapid growth of digital library or academic publications repository. This will lead to the tremendous growth in digital documents such as journals, articles, newspapers, magazines, e-books and etc. Indexing and categorization of those materials are crucial and these operations would be best performed automatically. This is where the language identifier is used in categorizing the documents at least according to their languages. So far, little research has been dedicated toward multi-lingual content. More so, there are not many researches dwelt on recognizing of Malay Documents.

The Malay language is spoken by over 300 million peoples and considered as the fourth largest language group after Chinese, English, and Hindi/Urdu [3, 25]. According to Dewan Bahasa dan Pustaka [1], there are 118 institutions throughout the world that teaching and pursuing research on Malay language. Although Malay has the same sentence structure (SVO) as English, Malay is a terse language and less pleonastic compared to English. Malay sentence can be easily remodeled on foreign structures such as English and Arabic. According to Azhar, Malay is different compared to English because Malay is a context-dependent language [2]. Similarly, Knowles et al. points out that Malay is syntactically different from English [10]. In another research, it is found that from approximately 30,000 Malay sentences, there were in average 21 words per sentence [2, 10, 14, 17]. Besides that, the Malay language consists of seven types of word formation, which are, affixation, reduplication, compounding, blending, clipping, acronyms and borrowing.

This research addresses implementation of a tool named Malay Language Document Identifier (MLDI) to recognize Malay and non-Malay digital documents in various forms. The organization of this paper is as follows: Section 2 presents some related works on Language Recognition using Artificial Neural Network. Section 3 describes the tool MLDI, the

implemented architecture, interface and language model creation. Section 4 shows the evaluation results and the last section gives an outlook to future work.

2 Related Works

The use of Internet in many facet of lives nowadays have created gigantic information highways that consist of millions even trillions of documents in various format used by the society in various languages [23]. Automatic language identifiers (AutoLId) are use to automate the process of indexing and categorization of these documents in diverse languages. The need of this AutoLId can benefit various domains that require a classifier to separate documents automatically in faster manner such as digital library, search engine or an automatic online translator service portals [15].

Accessing web pages is possible with the Internet search engines, which automatically crawl and index pages. Internet search engines were to appear in 1994 when the number of HTTP resources increased [24] and it is before the emergence and growth of the Web. The first pre-Web search engine was Archie, which performed keyword searches and retrieved names of files available via FTP [19]. Following this is the development of the first robot and search engine of the Web named Wandex by Matthew Gray in 1993 [29].

The growth of the Web has lead to the emergence of hundreds of search engines with different features. Primary search engines were designed based on traditional information retrieval methods that simply retrieves results from indexed databases and show the pages based on keyword occurrence and proximity. Though these traditional indexing models were found to be successful, it was revealed that these methods are not sufficient for a tremendously unstructured information resource such as the Web. Unfortunately, the completeness of the index is not the only factor in deciding the quality of the search results. Google introduces an innovative ranking system for the entire Web in order to increase the quality of the search results as described by Brin & Page [4]. Other efforts have been made to customize and specialize search tools.

There exist three generation of search engine technology described by Guozhen and friends [5], as discussed below.

- 1st generation: based on classic IR textual analysis techniques – retrieve documents upon analyzing the location and frequency of words in web pages to determine ranking using classic IR techniques such as TF*IDF, independent of each other.
- 2nd generation: Link analysis and ranking - Link analysis is also named as Web structure analysis. This technique means that the page link to another page and the latter is thought to be related and of high quality by the first page. By link analysis, law and trends of the web can mine out. The quality of each page relies on the global topology not on its neighboring pages. This global mutual dependence of page quality is computed through iteration. This technique has improves search quality and widely used in Google.
- The third generation: intelligent search engine - a search engine consists of intelligence features with friendly user interface which would provide nature way of interaction with the system such as ability to query with nature language, conceptual retrieval beyond literal matching and auto distinguish among multiple meanings of a user.

The architecture of search engines commonly consists of three component parts; web crawler, document indexing, searchers and result ranker. The main task of a web crawler is the automatic gathering of Web documents, which are usually stored locally. This automatic web document collectors are part of Web robots, which are applications that systematically traverse the Internet to perform some specific task. More specifically, it is commonly referred to as Web crawlers or Web spiders with numerous tasks namely Web survey, site checking and maintenance, site mirroring, and resource discovery [21].

Crawlers are designed for different purposes and can be divided into two major categories as mentioned by Suel and co-workers [27]. High performance crawlers form the first category with the goal to increase the performance of crawling by downloading as many documents as possible in a certain time. They use simplest algorithms such as Breadth First Search (BFS) to reduce running overhead. In contrast, the latter tries to maximize the benefit obtained per downloaded page. Crawlers in this category are generally known as focused Crawlers. Their goal is to find pages of interest using the lowest possible

bandwidth [27] with specific topic subjects such as scientific articles, pages in a particular language, mp3 files and images and was introduced in 1999 [22].

In principle there are two different techniques in implementing the automatic recognition language of a text document as mentioned by Grefenstette et al. [6], which are, word-based language recognition and the N-gram based identification [31]. Indexing methods are usually language dependent because they would need knowledge about the morphology of a language [7]. Partial morphological analysis of tokens as potential lexical or morphological units is done with the help of dictionaries of words and subwords typical for the language in question, including word-initial, word internal and word fragments used by Hisham El-Shishiny and friends [9]. This technique applies scoring scheme where each word fragments have positive or negative score and based on individual scores of detected features such as words, word prefixes, word suffixes and unigrams representing the characters used in a language and finally computed to be as the total score for each language. The highest score indicates the language of the text.

Meanwhile, Artificial Neural Networks (ANN) and rapid technological advancements have been applied to many applications because of their fascinating features, such as learning, generalizing, fast real-time computation and their modeling and classification capabilities. Because of these features, language identification processes are suitable for this type of application. The development of this language identification was achieved with the help of ANN using frequency analysis of letters [23]. There are many types of neural networks for various applications [7, 26, 28]. Multilayered perceptron neural networks (MLPNNs) are the simplest arch and most commonly used neural network [7]. Some learning algorithms such as Levenberg-Marquardt (LM) and Backpropagation with momentum (BPM) [23] could train MLPNN.

In another research, vector-space based category for high-precision is used to implement language recognition. The accuracy of this method depends on the size of the input document, the set of languages under consideration, and the features used. Linguini vector space could identify the language of documents as short as 5 to 10% of the size of average Web documents with 100% accuracy [20].

The basic idea of Linguini is to compute a similarity measure, based on cosine distance in feature-space. In the text retrieval, queries are compared with documents, while in categorization documents are compared with categories, or with other documents in a k-Nearest-Neighbor algorithm [13, 20]. Linguini was tested on 13 languages of European subset and each of them a collection of 100 Kbytes worth of text was gathered from the Internet. There are several other AutoLId techniques for document recognition such as string kernels [11], search engines, web-crawler and text words through visual discriminating features [8, 12, 16, 18, 30].

3 Approach and Method

There are two major approaches that was used in this research namely i) feature extraction and ii) implementation of BPNN for classification and recognition.

3.1 Feature Extraction

The architecture of MLDI consists of four components that are i) MLDI Program, ii) Matching Engine, iii) BPNN Engine and v) the User Interface as shown in figure 1. There are two databases: Malay words and Patterns & Training database. Malay words database consist of a large entry of words and its binary representation. While Patterns & Training database reside binary patterns resulted from pre-processing phase besides the weight and bias values gathered from the training session.

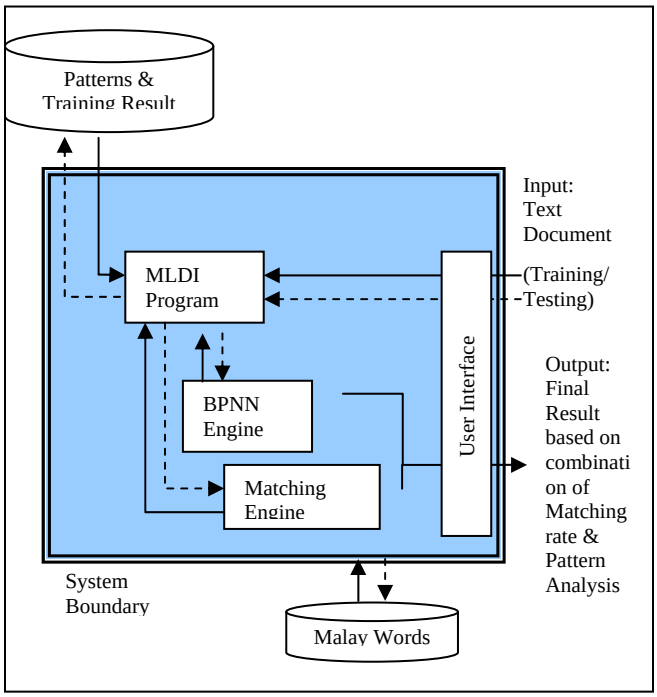

Figure 1 The architecture of MLDI of Malay written text prototype.

Documents from a url addresses or a specified file are passed thru the filtering processor which would eliminate and stripped all links, tags, images and symbols from the document The result consists of only text string as shown in figure 2.

Figure 2 Text extraction process

The extracted text are then split into sentences by locating all dots "." occurrences. Subsequently, all unwanted words and symbols would be removed from each of the sentences. Next, each filtered sentence would be further processed into individual words by removing the empty spaces as illustrated in figure 3. Once the word is extracted from the sentences, it would then be converted into small capital letters to be matched with the Malay words database. The matching words would then fetch its corresponding binary value and concatenated into a string of binary pattern. For an example, if the type is 'Kata Nama' then the returned result will be '0001', as shown in table 1. This binary string would be concatenated with previous string and finally a string of pattern representing a sentence would be extracted.

Table 1 Malay phrasal categories

Phrasal Category	Binary Value	Phrasal Category	Binary Value
Kata Nama	0001	Kata Keterangan	0110
Adjektif	0010	Kata Ganti	0111
Kata Kerja	0011	Kata Bantu	1000
Kata Seruan	0100	Kata Sendi Nama	1001
Kata Hubung	0101	Lain-Lain	1010

Besides producing the binary phrasal patterns, the Matching engine will also save any unmatched words and calculate matching score that will be used in the testing phase as shown in figure 3. The matching rate is based on number of words matched over the total of words exists in the current document and all non-matching words if any would be grouped together and displayed in the Main window of the user interface.

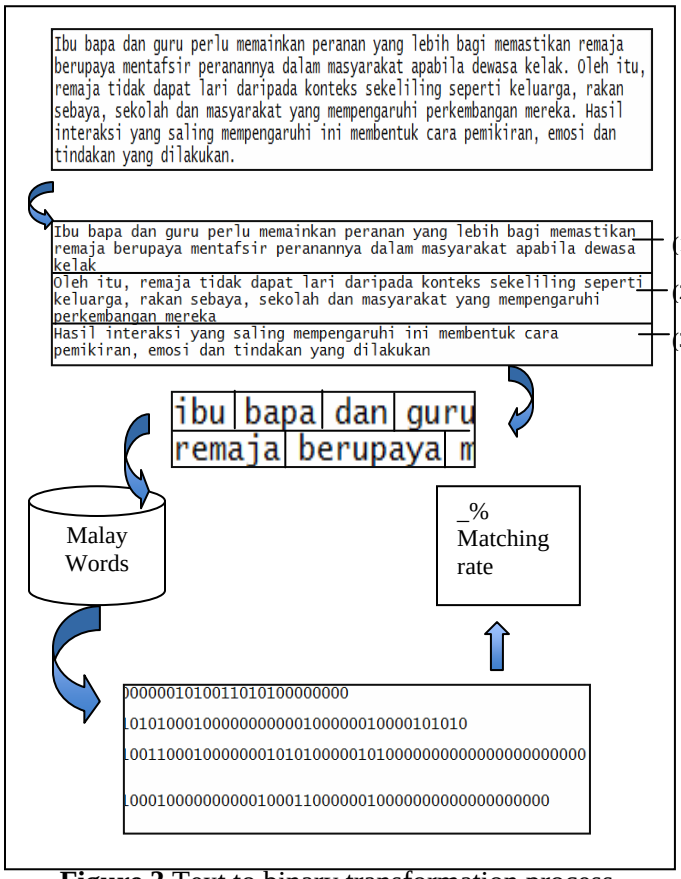

Figure 3 Text to binary transformation process

The generated binary patterns and its type would then be stored in the Pattern & training database. The binary patterns are normalized into an identified maximum length of the binary patterns. If the pattern were shorter than the length defined, the pattern would be concatenated with a sequence of set "0000", otherwise discarded as described in figure 4. The next stage is the classification and recognition process using BPNN.

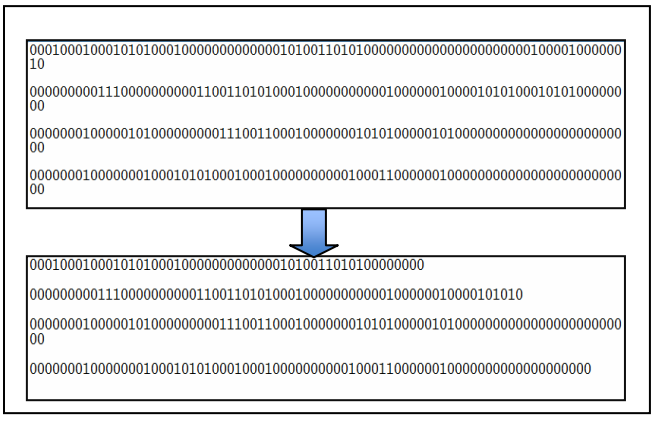

Figure 4 Data Normalization

Figure 5 illustrates the overall flow of MDLI from raw document which could be in Adobe PDF, MS Word, HTML web pages or other text document format.

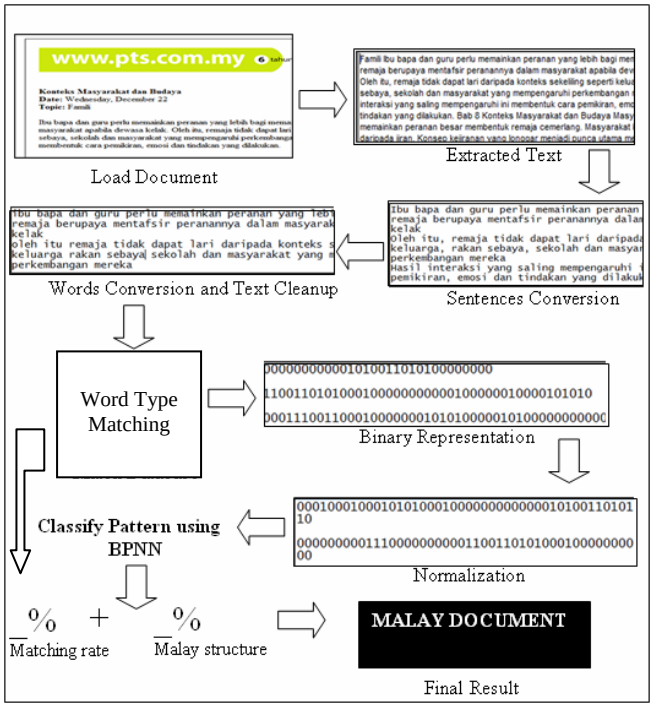

Figure 5 Flow process of MLDI prototype

3.2 BPNN

The system architecture developed is shown in figure 6. It consists of the BPNN architecture in learning and analyzing of Malays or non-Malay sentences. The training phase involved two types of data, which are Malay and non-Malay documents. Each type consists of 150 sentences with the total of 300 sentences for both types. The tool has the flexibility of altering sentences within the input parameter at the application Main window. A sentence of 24 words was determined as the shortest input parameter thru observation and confirmed by language expert. Meanwhile the longest length in the sentences depends on the training data available. Every word will be represented in four neurons thus make up the total neurons in the input layer as illustrated in equation 1 below.

$$N = \sum_{w=24}^{n} w \times 4 \quad (1)$$

Where N is the total number of neurons, w is number of words.

The BPNN consists of for example minimum of 96 adjustable input nodes, 19 adjustable number of hidden nodes using the ratio of 1/5 number of input nodes and 1 output node represent in the set of {0,1} which 0 represented as Malay sentences and 1 would be non-Malay sentences. The training result would be stored in the Pattern & Training database.

In the testing phase, the weights and biases values according to the length of the sentence being tested are retrieved from the Pattern & Training database. Finally, the result from the BPNN engine and the matching score from the matching engine are evaluated to produce the classification and recognition of a Malay document.

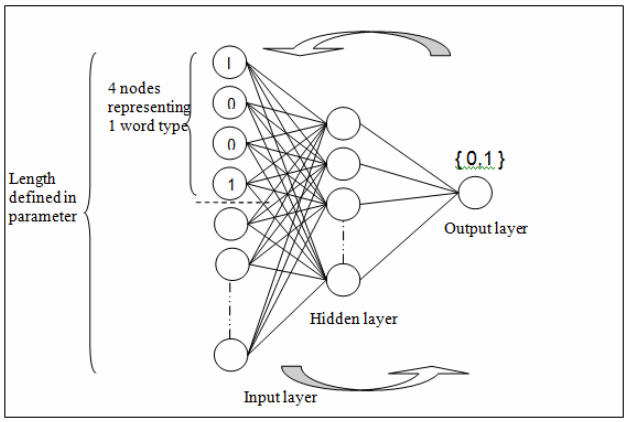

Figure 6 MLDI architecture

3.3 MLDI Interface Design

The MLDI graphical user interface build using GUI Visual Studio 2008 toolkit. Three main application windows with specific functions and processes are constructed. These application windows are as described below.

- The first application window is where the raw documents are identified and selected to be process, extract, filter for any unwanted words or symbols. The documents are than split into sentences and words to match the words in the database, which would identify the pattern according to the type of word. Next, normalization of all the sentences into the same length given by the processing

parameters. Finally, test the normalize sentence using BPNN and subsequently, show unmatched words, the recognition rate in the output window as shown in figure 7.

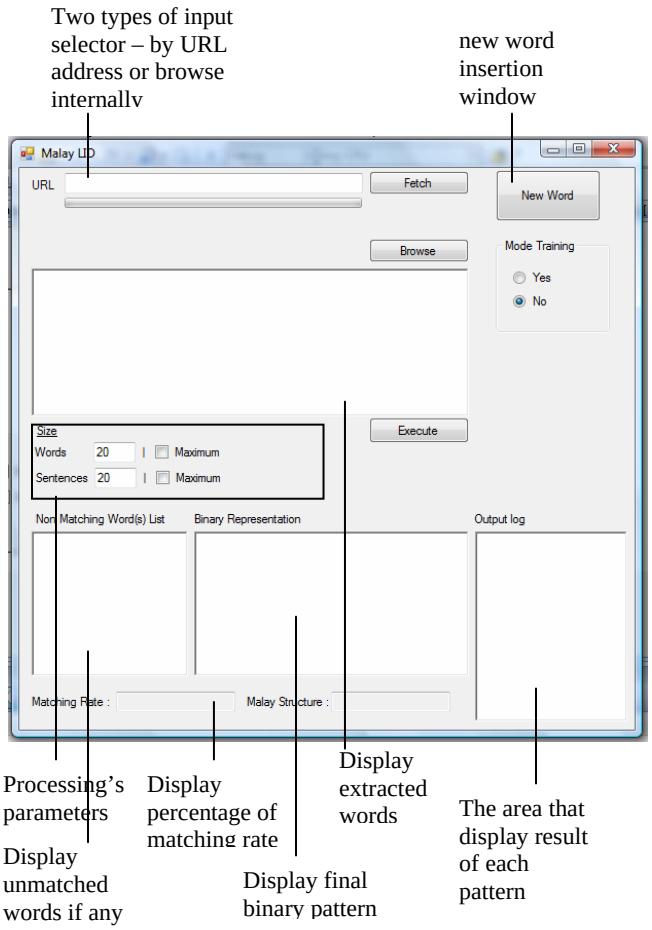

Figure 7 Application Main window

- This window shows the process of training using BPNN and called the training mode window. The default value of the training mode is 0, which means no training visible. However, if the user activated the training mode to 1 or yes, the second application window will appear with option of new input to choose from. These inputs can be extracted from the type selection lists, which consist of Malay or Non-Malay input. Once selection is made the BPNN training processes will take place. This function allows expansion of data types and flexibility in expanding the tool to be used for multi-lingual documents identification systems.

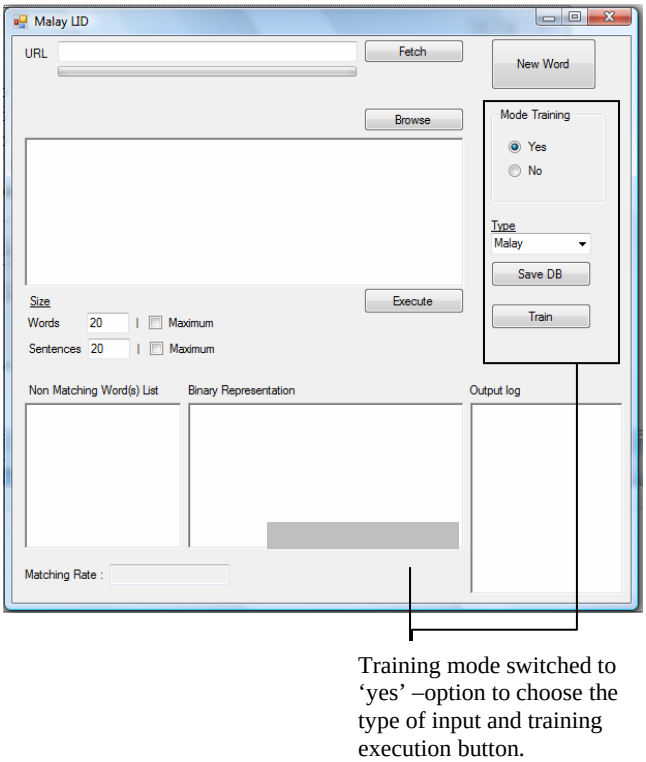

Figure 8 Application Main window (Training Mode)

- This window is for the expansion of the database. By pressing the New Word button, user are allowed to add new word in the existing Malay database. A textbox will appear for users to make their entry and select the word type from the selection list. Once selected, users are required to submit their entry by clicking on the Submit button.

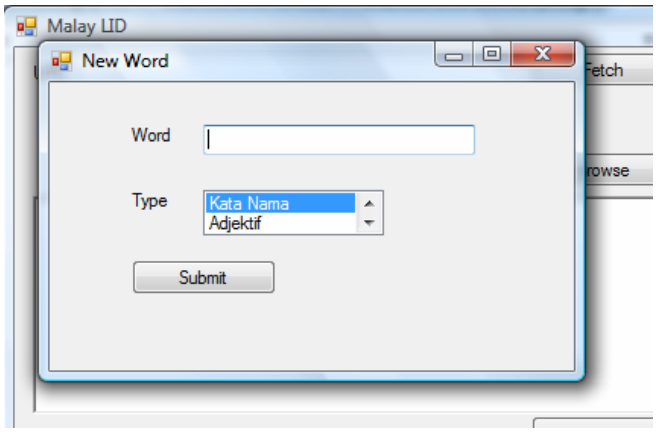

Figure 9 New Word Insertion window

3.4 The BPNN Training window

The BPNN Training window will appear once user clicked Training execution button on the second application window or training mode window. This window provides functions and processes such as setting of training parameters, Start button, Save button and Output area. The training parameters needed are numbers of input nodes (for instance 4 nodes representing 1 word), no of patterns (no of sentences), no of epochs, learning rate and threshold value.

The output area display the current Mean Square Error, Error Information Term and epoch number for every each process until the threshold or epoch value is met. Once the result is found and satisfied then the Save button action will save its training weights, biases and no of input nodes as new results to be use in the future testing phase.

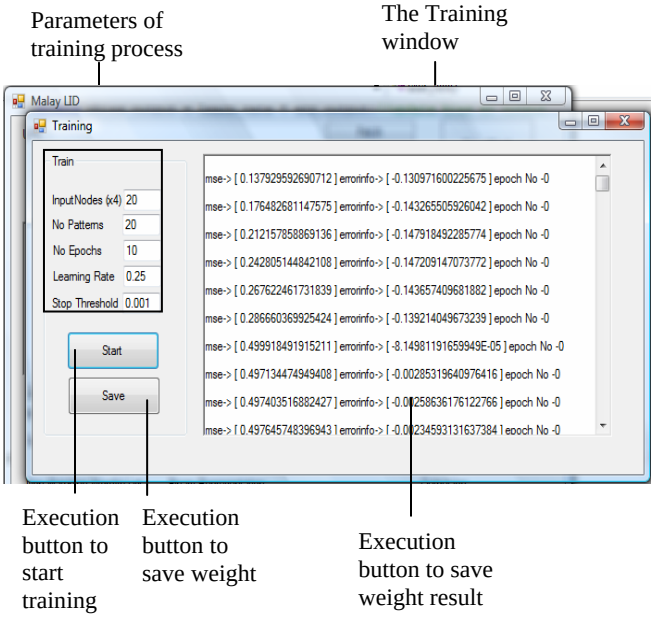

Figure 10 BPNN Training Window

4 Experiments and Results

The prototype is developed using C# as its programming language and MSSQL2008 for the database design. In the absence of a standard database, a Malay words database was prepared consists of 14,500 Malay words categorized into 9 groups which are *Kata Nama, Kata Kerja, Adjektif, Kata Hubung, Kata Sendi Nama, Kata Seruan, Kata Keterangan, Kata Bantu and Kata Ganti*. The rest of the other words which did not fall into these groups are combined together into one group called *Lain-Lain*.

Choice of the size of document patterns depends indirectly on the strength of feature extraction and classification techniques adopted. Therefore, in ensuring the success of the training process for better accuracy, the data collected are from a well known source of undeniable quality of publishing, has limited or none spelling errors, less graphical or multimedia presentation. Most of the data are from a collection of online newspapers and journal articles. Table 2 describes the input data for training and testing.

Table 2 Training and Testing Input Data

Language	Training Input	Testing Input	Sentence	Word
Malay	15	10	10	24
Non-Malay	15		10	24
Total	30	10	20	64

The training phase involved two types of data, which are Malay and non-Malay. Each type consisted of 150 sentences and makes up a total for 300 sentences used for training. The amount of sentences can be adjusted using an input parameter in the Main window. A sentence consists of 24 words determined by the input parameter or any longest length found in the sentences and every word will be represented in four neurons. Thus the total number of neurons in the input layer

given by total of words multiplies by four binary representations. Meanwhile the testing phase consists of 10 sentences of both Malay and non-Malay documents. The network is trained with a simple feed forward BPNN training algorithm, with momentum and a variable learning rate. In this experiment, a threshold is fixed to accept the obtained result at 0.01. The results obtained are tabulated in table 3.

Table 3 The results of document recognition by BPNN.

No. of documents	No. of sentences	Recognized as		Errors
		Malay	Non-Malay	
10	100	50	50	01

In this stage, the completed prototype was evaluated and analyzed in order to measure overall processing efficiency and accuracy. The performance comparison of the network is made in terms of recognition rate, classification accuracy and matching rate.

In classifying the document, an assumption is made that more than 50% must be identified as Malay structures to conclude that the file is a Malay language file. Table 4 shows the classification result for both BPNN training. From the result of BPNN, we can conclude that 90% classification accuracy was achieved.

Table 4 Classification rate

No.	Document	Classification Result: Malay structures	Conclusion: > 50% is Malay files	Classification correctness
1	Malay	80 %	Yes	Correct
2	Malay	60 %	Yes	Correct
3	Malay	70 %	Yes	Correct
4	Malay	60 %	Yes	Correct
5	Malay	60 %	Yes	Correct
6	Non-Malay	40 %	No	Correct
7	Non-Malay	20 %	No	Correct
8	Non-Malay	60 %	Yes	Incorrect
9	Non-Malay	0 %	No	Correct
10	Non-Malay	50 %	No	Correct

While, table 5 shows the matching results where 100% of the documents are classified accurately. Thus combination of both tests will generate more precise recognition.

Table 5 Matching rates

No.	Document	Number of sentences	Matching Rates	Conclusion: > 50% is Malay files
1	Malay	10	90 %	Yes
2	Malay	10	91.41 %	Yes
3	Malay	10	81.35 %	Yes
4	Malay	10	89.05 %	Yes
5	Malay	10	86.6 %	Yes
6	Non-Malay	10	7.3 %	No
7	Non-Malay	10	3.86 %	No
8	Non-Malay	10	8.2 %	No
9	Non-Malay	10	6.67 %	No
10	Non-Malay	10	5.21 %	No

Table 6 shows that the combination results of both matching and classification rate give 100% correctness in classifying between Malay and Non-Malay files. Referring to table 4, document number 8 was concluded to be a Malay document since its classification results presented 60% accuracy. However, the result concluded from table 5 illustrates a matching Malay word of only 8.2%. Thus this could means that document 8 has the right structure but mostly written in a different language. Therefore, it is concluded that document 8 could be a non-Malay document.

Table 6 combination results of matching and classification rate

No.	Document	Satisfied Malay Matching Rates	Satisfied Malay structure rates	Conclusion
1	Malay	Yes	Yes	Malay
2	Malay	Yes	Yes	Malay
3	Malay	Yes	Yes	Malay
4	Malay	Yes	Yes	Malay
5	Malay	Yes	Yes	Malay
6	Non-Malay	No	No	Non-Malay
7	Non-Malay	No	No	Non-Malay
8	Non-Malay	No	Yes	Non-Malay
9	Non-Malay	No	No	Non-Malay
10	Non-Malay	No	No	Non-Malay

5 Conclusion and Future Works

A BPNN based recognition system with matching algorithm is proposed for Malay document analysis

and recognition. There have not been many research on Malay Language Document Identifier. Therefore this paper addresses the needs and development of a tool that can recgnize and analyze malay documents. This research has proven that BPNN can learn to recognize Malay sentences' structure with a reasonable classification rate. However, the combination of Malay words matching rate and results from BPNN testing phase gives a better conclusion rate. The results obtained demonstrate the effectiveness of both methods. This research is hope to generate more interest in Malay language technology using Neural Network approach. It could also be used as a component in Malay language Web-focused crawler, as it helps in identifying the language of documents.

In the future, the size of the database could be extended to include all possible phrasal of Malay language. Besides that, it can also be implemented for different languages by replacing the database and the dictionary used. Besides that, this research can utilize morphological analyser of the Malay language.

References-:

[1] Asmah, O., Morfologi-sintaksis Bahasa Malayu (Malay) dan Bahasa Indonesia: Satu Perbandingan Pola, *Dewan Bahasa dan Pustaka, Kuala Lumpur,* 1968.

[2] Azhar, M.S., Discourse-Syntax of "YANG" In Malay (Bahasa Malaysia), *Kuala Lumpur: Dewan Bahasa dan Pustaka*, 1988.

[3] Balisoamanandry, R., Computational Analysis of Affixed Words In Malay Language, *Universiti Sains Malaysia*, 2001.

[4] Brin, S. & Page, L.. The anatomy of a large-scale hypertextual websearch engine. *Proceedings of the 7th International WWW Conference,Brisbane, Australia*, pp 107-117, 1998.

[5] Guozhen, Feng, Xueqi, Cheng, Shuo, Bai, SAInSE: An Intelligent Search Engine Based on WWW Structure Analysis, *IEEE explore*, pp 2, 2001.

[6] Grefenstette, G., Comparing Two Language Identification Schemes, *JADT 3rd International conference on Statistical Analysis of Textual Data*, 1995.

[7] Haykin, S., Neural Networks: A Comprehensive Foundation, *Macmillan College Publishing Comp*, 1994.

[8] Hazen, T. J. & Zue, V. W., Segment-based automatic language identification. *The Journal of the Acoustical Society of America*, Volume 101, Issue 4, pp. 2323-2331, April 1997.

[9] Hisham E. S., Troussov, A., Takeuchi D. M. M., Nevidomsky, A. & Volkov, P., Arabic Language Resources and Tools Conference, *Cairo Egypt*, 2004.

[10] Knowles, G. O. & Don, Z. M., Word Class in Malay: A Corpus-Based Approach, *Kuala Lumpur: Dewan Bahasa dan Pustaka*, 2006.

[11] Kruengkrai, C., Srichaivattana, P., Sornlertlamvanich, V. & Isahara, H., Language Identification Based on String Kernels, *Proceedings of ISCI*, 2005.

[12] Lamel, L. F. & Gauvain, J., L., High performance speaker-independent phone recognition using CDHMM, *Third European Conference on Speech Communication and Technology*, pp. 121-124, 1993.

[13] Ling, C.X. and Wang, H., Computing Optimal Attribute Weight Settings for Nearest Neighbor Algorithms, *Artificial Intelligence Review*, Vol. 11, pp. 255-272, 1997.

[14] M.Blanche, L., Sentence Analysis In Modern Malay, Cambridge, *At The University Press, London*, 1969.

[15] Muthusamy, Y., K., Bamard, E. & Cole, R. A., Reviewing Automatic Language Identification, *IEEE Signal Processing Magazine*, October 1994.

[16] Naghabhushan, P. & Pai, R., M., Modified Region Decomposition Method and Optimal Depth Decision Tree in the Recognition of non-uniform sized characters – An Experimentation with Kannada Characters, *Journal of Pattern Recognition Letters*, vol. 20, pp. 1467-1475, 1999.

[17] Omar, A., Morfologi-sintaksis Bahasa Malayu (Malay) dan Bahasa Indonesia: Satu Perbandingan Pola, *Dewan Bahasa dan Pustaka, Kuala Lumpur*, 1998.

[18] Padma, M.C. & Vijaya, P.A., Language Identification of Kannada, Hindi and English Text Words Through Visual Discriminating Feature, *International Journal of Computational Intelligence Systems*, Vol.1, No. 2, pp. 116–126, May, 2008.

[19] Poulter, A., The design of World Wide Web search engines: a critical review, *Program*, 31(2), pp 131-145, 1997.

[20] Prager, J. M., Linguini: language identification for multilingual documents, *Journal of*

Management Information Systems, Volume 16, Issue 3, pp. 71 – 101, 1999.

[21] S. da Silva, Altigran, Eveline A. Veloso Paulo B. Golgher, Berthier Ribeiro-Neto Alberto H. F. Laender Nivio Ziviani, CoBWeb – A Crawler for the BrazilianWeb, *Department of Computer Science Federal University of Minas Gerais*, 2000.

[22]S. Chakrabarti, M. van den Berg & B. Dom, Focused crawling: A new approach to topic specific resource discovery. *Proceedings of the Eighth World Wide Web Conference*, pp 545–562, 1999.

[23]Sagiroglu, S., Yavanoglu U. & Guven, E. N., Web based Machine Learning for Language Identification and Translation, *Sixth International Conference on Machine Learning and Applications*, *IEEE*, 2007.

[24]Schwartz, C.. Web search engines. *Journal of the American Society for Information Science*, 49(11), pp 973-982, 1998.

[25]Shin, Y. H., Malay/Indonesian for an Official Language of the East Asian Community, Retrieved http://www.arenaonline.org/content/view/305/151/, Jan.30, 2005.

[26]Shuzlina Abdul Rahman, Izham Fariz Ahmad Jinan, Ku Shairah Jazahanim, Azlinah Mohamed, Intelligent Water Dispersal Controller Using Mamdani Approach, *Lecture Notes In Computational Intelligence*, WSEAS Press, pg. 154-159, June 2007.

[27]Suel, T. & Shkapenyuk, V., Design and Implementation of a High-Performance Distributed Web Crawler, *In Proceedings of the 18th International Conference on Data Engineering (ICDE'02)*, San Jose, CA., 2002.

[28]Varela, A., Moreno, H. G. & Lopez C. P., Prediction of the Effect of Premium Changes on Companies Insurance Customer Rates Using Neural Networks, *WSEAS Transactions on Information Science and Applications*, Issues 4, Vol 3, August 2006.

[29]Wall, A. (2004). History of search engines & web history from http://www.search-marketing.info/search-engine-history/, Retrieved March 28, 2008.

[30]Yi Hu, Ruzhan Lu, Yuquan Chen, A Way for Applying Conceptual Knowledge Derived from a Traditional Dictionary to Text Clustering, *WSEAS Transactions on Information Science and Applications*, Issues 8, Vol 4, August 2007.

[31]Zissman, M.A. & Singer, E, Automatic language identification of telephone speech messages using phoneme recognition and N-gram modeling, Acoustics, *Speech, and Signal Processing, 1994 ICASSP-94, 1994 IEEE International Conference*, Vol. 1, pp. 305-308, 1994.