Soft Computing Approaches for tagging Arabic text

This paper was downloaded from TechRxiv (<https://www.techrxiv.org>).

LICENSE

CC BY 4.0

SUBMISSION DATE / POSTED DATE

03-09-2021 / 09-09-2021

CITATION

Yousif, Jabar (2021): Soft Computing Approaches for tagging Arabic text. TechRxiv. Preprint.
https://www.techrxiv.org/articles/preprint/Soft_Computing_Approaches_for_tagging_Arabic_text/16569420

DOI

[10.36227/techrxiv.16569420.v1](https://doi.org/10.36227/techrxiv.16569420.v1)

Soft Computing Approaches for tagging Arabic text

Jabar H. Yousif

Faculty of Computing and Information Technology
Sohar University, P.O.Box.44, P.C.311, Sohar, Sultanate of Oman
E-mail: jyousif@soharuni.edu.om

ABSTRACT

This paper discusses the application of soft computing (SC) techniques in natural language processing. The current research proposes the design and implementation of automatic models that distinguish the text in the Arabic language and then distinguish it into the essential parts of speech. Speech tag characterization (POS) is the process of classifying words in a sentence based on their meaning, functions, and types (noun, verb, adjective, etc.). This work developed two-stage labeling models. The first stage is to process the text before execution by removing the suffixes attached to it. And then the second stage is to design models by applying mathematical models using Multilayer Perceptron (MPL), Full Recurrent Neural Network (FRNN), and Support Vector Machines (SVM). The current system helps classify words and allocate the correct parts of speech according to their wholesale position. To test the effectiveness of the proposed models using two different languages (Arabic and Hindi). The results showed the effectiveness of the proposed models is successfully solving the problem of clarification of words for the Arabic text. Also, compared to previous studies, the proposed models achieved high accuracy by classifying the parts of speech with an accuracy of up to (99%).

Keywords: Information Systems, Soft Computing, Artificial Neural Networks, Arabic Text Processing Part of Speech Tagger

1. INTRODUCTION

Soft Computing (SC) is a collection of computational models, which utilizes tolerance for imprecision, uncertainty, robustness, and tiny solution cost to formularize real-world problems. SC generally includes Artificial Neural Networks (ANN), Fuzzy Logic (FL), Evolutionary Computing, Genetic Algorithms (GA), and Rough Set Theory [30,32]. The Main characteristics of SC are their ability to evaluate, decide, check, and calculate within a vague and imprecise domain, emulating the human skills in the execution to learn from experience. Natural Language Processing (NLP) can be defined

as an automatic or semi-automatic approach to processing the human language [17, 25, 33]. Recently, Arabic and Hindi languages processing has become a primary focus of research and commercial development. Most NLP applications often include speed and accurate POS tagger as one of its central core components [21]. The Part of Speech (POS) is a classification of words according to their meanings and functions. The POS tagger plays a crucial and essential role for most NLP applications such as machine translation, information extraction, speech recognition, and grammar and spelling checkers [13].

Moreover, the accuracy of the POS tagging is determined by factors like ambiguous words, phrases, unknown words, and multipart words. Specific features excite scientists to espouse neural network-based solutions in solving problems [12, 15]. The most interesting features of NN include massive parallelism, uniformity, generalization ability, distribution representation and computation, learnability, trainability, and adaptation. Neural approaches have been performed successfully in many aspects of artificial intelligence, such as image processing, NLP, speech recognition, pattern recognition, and classification tasks [2]. The Recurrent Neural Networks (RNN) consists of neurons with feedback connections, which are biologically more plausible. And computationally more powerful than other adaptive models like Hidden Markov Models (HMM) [31], Feed-Forward Networks and Support Vector Machines (SVM) [14,16, 24, 34]. The SVMs are considered a supervised learning method that is used to perform binary classification and regression tasks. They belong to a family of generalized linear classifiers. The main advantages of SVM are that they simultaneously minimize the experimental classification error and maximize the geometric margin [6,12].

2. THE DIFFERENCES BETWEEN POS TAGGER MODELS

Part-of-speech tagging is a complicated process, not just having a list of words and their parts of speech as, at times, some words can represent more than one part of speech. Some can be in the form of ambiguous phrases [4, 5]. Hence, it is hard to build a POS tagger that can tag with an accuracy of

100 percent for extensive training data. Typically, different approaches have been implemented to address the part of speech tagger such as the rule-based [9, 18 and 19], stochastic [7, 10 and 11], neural network [1, 12, 13, 14, 15, 16, 20, 22, 23 and 26] or the hybrid systems [8]. The rule-based and stochastic approaches need a vast amount of data to adapt and implement the POS tagger. It has been known that the neural methods only use a small amount of data to perform the training and learning stages.

Moreover, the neural-based approaches are not only consummate the associations (word-to-tag mappings) from a representative training data set, but they can also be generalized to the unseen [1, 17]. Overall, several advantages of stochastic taggers can be identified over the rule-based taggers. They avoid the need for diligent manual rule building and probably obtain helpful information that humans may overlook. However, these probabilistically driven ones have the disadvantage in which the linguistic input is only captured indirectly in large statistics tables. In contrast, the rule-based taggers need the minimum storage requirements and, at the same time, are more portable [17].

3. PERFORMANCE MEASURES

The performance evaluation of classification process is a crucial matter in the machine learning systems, because it is unfeasible to contrast learning algorithms or even know whether a hypothesis should be used. The most important attribute in the assessment of a part-of-speech tagger is accuracy [28]. Thus, the quality of the output depends on the comparability of conditions [17] such as:

****Tag-set size:** Normally, using a small number of tag-set can help to give high accurate tagging but it does not offer as much information or disambiguation between the lemmas as a larger one would.

****The corpus type:** A corpus (corpora is the plural) is a set of text that collected for a purpose. The type of corpus affects the quality of taggers output when the genre or type of the corpus data differs from the tagged material.

****Vocabulary type:** the tagging of specific texts such as the medical or legal texts requires a training corpus that has examples of such texts; otherwise, the unknown words will be unnaturally high). Likewise, the high instance of idiomatic expressions in the literary texts often leads to inaccuracy. However, the ambiguous words and phrases, unknown words and multi-part words can affect the accuracy of POS tagging. Ambiguity appears at different levels of the language processing sequence such as syntax or semantic phase [28].

Figure1 : Natural Language P process ing applications

4. RELATED WORK

This section is presented a survey of previous work in fields of part of speech tagger using of rule based, stochastic and neural networks. **Schmid [26]** established a Net-Tagger which trained on a large corpus called Penn Treebank. This tagger has a context window of three preceding words and two succeeding words The corpus performed significantly and compared to statistical approaches based on “Trigram model” and “Hidden Markov model” (HMM). **Diab et al. [10]** they utilized a SVM approach

toward automatically tagging POS and annotate base phrases (BPs) in Arabic text. She attained score of 99.12 for tokenizing when $F_{\beta}=1$, and score of tagging accuracy is 95.49%. While, recorded score of 92.08 for chunking when $F_{\beta}=1$. **Pérez-Ortiz [23]** implemented a Discrete-time Recurrent Neural Networks (DTRNN) tagger to tag ambiguous words from the sequential information stored in the network's state. The experiments computed the error rates when tagging text taken from the Penn Treebank corpus. **Ahmed [1]** used MLP-tagger with three-layers using error back-propagation learning algorithm. The tagger was implemented on SUSANNE English tagged-corpus consisting of 156,622 words. The MLP-tagger is trained using 85% of the corpus. Based on the tag mappings learned, the MLP-tagger demonstrated an accuracy of 90.04% on test data that also included words unseen during the training. **Jabar [23]** implemented an Arabic part of speech based multilayered perceptron. The experiments evinced that the MLP tagger has high accurate (of 98%), with low training time and fast words tagging. they used a little amount of data to achieve the adaptation and learning of network. **Jabar [12]** proposed Arabic part of speech based support vectors machine. The radial basis function is used as a linear function approximation. The experiments evinced that the SVM tagger has a high accuracy and recall about (99.99%). **Jabar [14]** proposed an Arabic part of speech based Fully Recurrent Neural Networks (FRNN). The back-propagation through time (BPTT) learning algorithm is used to adjust the weight of the network and associate inputs to cyclic outputs. In order to accurately predict the syntactic classification tagging, an encoding criteria is also presented and performed. The experiments evinced that the FRNN tagger is accurate and achieved 94% in classification phase. Similarly, the POS disambiguation problem was successfully solved. **Khoja [19]** proposed the APT Arabic Part-of-Speech Tagger which used a combination of both the statistical and rule-based techniques. The APT tag-set are derived from the BNC English tag-set, which was modified with some concepts from the traditional Arabic grammar.

5. TAGGERS DESIGN

The proposed system is consisting of two main stages as depicted in Figure 2. The key function of first stage is to prepare the input data sets for next stage. This stage is written and implemented using VBA commands for Excel. While, the main function of second stage is to implement the automatic taggers. These taggers are designed and implemented using NeuroSolutions for Excel software. The first stage is called “pre-processing phases”[17]. It is implemented and utilized to achieve the following tasks: Text Normalization, Text Tokenization and Text Encoding.

The Text normalization is used to convert the input text from free text into suitable forms to be used in next stage. In general, the input text can be configured either into a text file or XML file. Therefore, the system is designed to disregard all the HTML tags and extract the pure contents of the document. Then, the text tokenization is distributed the pure text into simple tokens such as numbers, punctuation, symbols, and words. An algorithm has been developed to implement and perform the text tokenization task. Lastly, the text encoding is performed to transform the input data into a suitable digital form, which the network can identify and use [13, 17].

The proposed encoding method aims to solve the drawbacks of previous encoding schemes. Consequently, it aims to increase the number of significant positions and decrease the usage of memory storage.

Moreover, the proposed scheme uses new concepts like the probability of the word and the numerical values of each word. An algorithm is developed to compute the probability of each word. This algorithm is written and implemented using VBA commands for Excel. Finally, the numerical value of each word in the input based on the new encoding scheme is determined. Subsequently, every word in the sentence is associated with a bit-vector, i.e., the size of which is equal to the number of different grammatical categories (for parts of speech) in a specific language.

Figure 2: System Architecture

The second stage is “processing phase” which is used to design and implement automatic taggers which include the following tasks:

****The MLP tagger [13, 15 and 16]:** This task is designed to implement the POS tagger for Arabic text using the Multilayered Perceptron technique. The architecture of the network has one hidden layer, 23 processing elements tagged as input, and 23 processing elements tagged as output. The maximum number of epochs is 1000. The TanhAxon is implemented as a transfer function in the hidden and output layer. The TanhAxon applies a bias and Tanh function to each neuron in the layer.

****The Recurrent tagger [14]:** This task is designed to implement the POS tagger for Arabic text using the fully Recurrent Neural Networks technique. The tagger has one hidden Fully Synapse (The Synapse family implements the dynamic linear mapping characteristics of the neuron. A synapse

connects two layers of axons) with 230 processing elements as input and 23 processing elements as output. A static controller for a network with multi-channels tapped delay line memory structure (TDNNAxon) which has 10 tapes, usually used as the Input Axon. Likewise, in both the hidden and output layers, the TanhAxon as a transfer function is implemented. The RNN has 23 processing elements (Columns) tagged as Input and 23 processing elements (Rows) tagged as Output.

****The SVM tagger [12]:** This task is used to implement the POS tagger for Arabic text using the terminology of super vector machine algorithms. The SVM architecture have 23 PEs as input set x_i , 23 PEs as output set d_i and have no hidden layer. The maximum number of epochs is 1000 and set the step size to 0.01. The learning algorithm is based on the Adatron algorithm which is extended to the RBF network.

6. EXPERIMENTS AND RESULTS

The experiments undertaken are achieved using the Arabic tag-set which is proposed by Khoja [19]. The tag-set contains 177 tags that include various categories. The extraction of words into basic roots is not considered in this study. This study supposed that the words were segmented before POS tagging began. The experiments covered the three proposed taggers in this paper SVM tagger, MLP tagger and FRNN tagger. The input text is encoded into a suitable form and then it is divided into three categories; training data sets, cross validation data sets and test data sets. The Cross validation computes the error in a test data sets at the same time that the network is being trained with the training set. The Genetic Algorithm (GA) is used as a heuristic optimization in the problem of finding the best network parameters [27]. It establishes with an initial population of randomly created bit strings.

These initial samples are encoded and applied to the problem. The study under taken is used the GA methods for improving the learning rule parameters such as step size and momentum value [13, 14 and 17]. This will enable the optimization of the momentum values for all Gradient components in NeuroSolutions software that use momentum. Besides, it used to determine the number of processing elements. Likewise, to tolerate the enhanced fit specimen in the population to reproduce at a higher

rate is to use a selection method based on the roulette wheel selection technique. The standard method to assess the tagger performance is usually determined by the percentage of correct tag assignments.

The NeuroSolutions for Excel software is provided six methods to test the networks performance such as the mean squared error (MSE), the normalized mean squared error (NMSE), the correlation coefficient(r), the percent error, Akaike's information criterion (AIC) and Rissanen's minimum description length (MDL) criterion [17].

The NMSE is the estimation of the overall deviations between predicted values and measured by the network. It is defined as follows:

$$NMSE = \frac{\sum_{j=0}^P \sum_{i=0}^N (d_{ij} - y_{ij})^2}{\sum_{j=0}^P \frac{N \sum_{i=0}^N d_{ij}^2 - (\sum_{i=0}^N d_{ij})^2}{N}}$$

The MSE "mean squared error" is two times the average cost which is computed as follows:

$$MSE = \frac{\sum_{j=0}^P \sum_{i=0}^N (d_{ij} - y_{ij})^2}{NP}.$$

Where, P is the number of output processing elements.

N is the number of exemplars in the data set.

y_{ij} is the network output for the exemplar i at processing element j .

d_{ij} is the desired output for the exemplar i at processing element j .

The correlation coefficient (r) is the rate relation between a network output x and a desired output d. It is defined as follows:

Usually, the MSE "mean squared error" is used as evaluation function for the network output reliability. The best network results for training data of the proposed taggers (MLP, FRNN and SVM) reported a minimum and final mean squared error (MSE) as depicted in Tables 1, 2 and 3 accordingly. Moreover, Figures 4, 5, and 6 are illustrated the training results graph of proposed neural taggers MLP, FRNN and SVM respectively. Figure3 shows the comparison results of MSE of Proposed taggers. The MLP network recorded a final MSE of 0.000103638. And the FRNN network recorded

$$r = \frac{\sum_i (x_i - \bar{x})(d_i - \bar{d})}{\sqrt{\frac{\sum_i (d_i - \bar{d})^2}{N}} \sqrt{\frac{\sum_i (x_i - \bar{x})^2}{N}}}$$

a final MSE of 0.020686609. Lastly the SVM network recorded a final MSE of 0.000878348.

Table 1. MSE for MLP tagger

Best MLP	Training	Cross
Run NO.	2	2
Epoch NO.	1000	1000
Minimum MSE	0.000103638	0.000104718
Final MSE	0.000103638	0.000104718

Table 2. MSE for FRNN tagger

Best FRNN	Training	Cross Validation
Run NO.	3	3
Epoch NO.	1000	1000
Minimum MSE	0.020686609	0.02351733
Final MSE	0.020686609	0.02351733

Table 3. MSE for SVM tagger

Best SVM	Training	Cross
Run NO.	1	1
Epoch NO.	1000	1000
Minimum MSE	0.000878348	0.008483563
Final MSE	0.000878348	0.008483563

7. COMPARISON & CONCLUSIONS

7.1 Comparison with related work

The comparison study has to be implemented accurately

because the features used here to identify the languages and the tag sets are different from the previous studies [12]. The comparison of proposed taggers with other existing taggers is a complicated matter because the tagger accuracy relies on numerous parameters such as language difficulty (ambiguous words, ambiguous phrases), the language nature (English, Arabic, Hindi, Chinese, etc.), the training

data size, the tag-set size, and the evaluation measurement criteria [13, 14]. The Tag-set size has a significant impact on the tagging process. The proposed taggers are assessed using the measurement of accuracy, besides MSE aspects. In addition, the amount of data used in the training and learning stages is considered. A comparison study of proposed taggers with the results of other taggers explained that the proposed taggers achieved a high accuracy rate when using GA optimization techniques which improved the values of the momentum rate and the step size. The proposed taggers (SVM, MLP, and FRNN) achieved high accuracy of 99% at the last experiments when the GA optimization process is implemented. Table4 summarizes the comparison information with other researchers. Figure7 illustrates the overall comparison results.

7.2 Conclusions

The research mainly aims to implement an automatic and accurate tagging system that can be used as a central core component for NLP applications. The automatic part of the speech tagging system is deployed based on neural network techniques, which can tag the free texts automatically. The study demonstrated variant kinds of taggers that can solve the problem associate with the contraction of languages such as Arabic part of speech and Hindi part of speech. The new approaches are highly accurate with low processing time and high-speed tagging. Two stages of automatic tagging system-based SVM, MPL, and FRNN, are implemented and designed. The proposed system helps to classify words and assign the correct POS for each of them. The results are exceptionally encouraging, with correct assignments and recall of about 99%. The Genetic Algorithm is utilized to optimize the network variables like the momentum rate and step size. Also, the disambiguation of Arabic words is solved by the proposed Arabic POS taggers.

Figure3: MSE of Proposed taggers

8. FUTURE WORK

This paper presented the design and implementation of an automatic tagger that can tag a free text directly and combining each word with its correct part of speech. The work focuses only on two types of files (text files and HTML files). Therefore, it is preferable to include more files type. And, if it is possible to extract the text directly from the website, it will be very encouraging. The current work comprises two separate stages; the first pre-processing phase, implemented using VBA codes. Besides, the second stage is the processing phase which is utilized using the NeuroSolutions software. Advantageously, the steps are merged into one part to produce a portable system that can use with any other application.

REFERENCES

- [1]. Ahmed. "Application of Multilayer Perceptron Network for Tagging Parts-of-Speech", Proceedings of the Language Engineering Conference (LEC'02), IEEE, 2002.
- [2]. Aleksander, Igor, and Morton, Helen, An Introduction to Neural Computing, Chapman and Hall, London, 1990.
- [3]. Al-Sulaiti's Latifa. "Online corpus". <http://www.comp.leeds.ac.uk/latifa/research.htm>.
- [4]. Attia, M. " A large-scale computational processor of the Arabic morphology and applications", MSc. thesis, Dept. of Computer Engineering, faculty of Engineering, Cairo University, 2000.
- [5]. Beesley, K. "Finite-State Morphological Analysis and Generation of Arabic at Xerox Research: Status and Plans" , ACL, Arabic NLP Workshop, Toulouse, 2001.
- [6]. Ben-Hur, A., Horn, D., Siegelmann, H. T., and Vapnik, V. (2001), Support vector clustering, Journal of Machine Learning Research, 2:125–137.
- [7]. Brants, T. "TnT- a statistical part-of-speech tagger", proceedings of the 6th ANLP conference, Seattle, WA, 2000.
- [8]. Brill, E. "Unsupervised learning of disambiguation rules for part of speech tagging". Proceedings of third ACL Workshop on Very Large Corpora, 1995.

- [9]. Brill, E. "A simple rule-based part-of-speech tagger", proceedings of ANLP-92, 3rd Conference on Applied Natural Language Processing, pp 152–155, Trento, IT, 1992.
- [10]. Diab, M., Kadri H. & Daniel J. "Automatic tagging of Arabic text: from raw text to base phrase chunks", proceedings of HLT-NAACL-04, 2004.
- [11]. Gimenez, J. & Lluís M. "Fast and accurate part-of-speech tagging: The SVM approach revisited", proceedings of the International conference on recent advances on natural language processing, Borovets, Bulgaria, 2003.
- [12]. Jabar H. Yousif, & Sembok, T., "Arabic Part-Of-Speech Tagger Based Support Vectors Machines", proceedings of International Symposium on Information Technology, Kuala Lumpur Convention Centre, ISBN 978-1-4244-2328-6©IEEE, Malaysia, August 26-29, pp: 2084-2090, 2008. URL:http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4632066.
- [13]. Jabar H. Yousif, & Sembok, T., "Design And Implement An Automatic Neural Tagger Based Arabic Language For NLP Applications", Asian Journal of Information Technology Vol. 5, Issue 7, ISSN 1682-3915, pp 784-789, 2006. DOI: 10.3923/ajit.2006.784.789.
- [14]. Jabar H. Yousif, & Sembok, T., "Recurrent Neural Approach Based Arabic Part-Of-Speech Tagging", proceedings of International Conference on Computer and Communication Engineering (ICCCE'06), Vol 2, ISBN 983-43090-1-5© IEEE, KL-Malaysia, May 9-11, 2006.
- [15]. Jabar H. Yousif, & Sembok, T., "Arabic Part-Of-Speech Tagger Based Neural Networks", proceedings of International Arab Conference on Information Technology ACIT2005, ISSN 1812/0857. Jordan-Amman-2005.
- [16]. Jabar H. Yousif, and Dinesh Kumar Saini, "Hindi Part-Of-Speech Tagger Based Neural Networks", Journal of Computing, Volume 3, Issue 2, pp59-65 ISSN 2151-9617, NY, USA, February 2011.
- [17]. Jabar H. Yousif, "Information Technology Development", LAP LAMBERT Academic Publishing, Germany ISBN 9783844316704, 2011.
- [18]. Khoja S, Garside, R. & Gerry, K. "An Arabic tagset for the morphosyntactic tagging of Arabic", corpus linguistics, Lancaster University, Lancaster, UK, 2001.
- [19]. Khoja, S. "APT: Arabic part-of-speech tagger", proceedings of the student workshop at the second meeting of the north American chapter of the association for computational linguistics (NAACL2001), Carnegie Mellon University, Pennsylvania, 2001.
- [20]. Ma, Q., Uchimoto, K., Murata, M. & Isahara, H. "Elastic neural networks for part of speech tagging", proceedings of IJCNN'99, pp 2991–2996, Washington, DC, 1999.
- [21]. Mahtab, N. & Choukri, K. "Survey on Industrial needs for Language Resources", 2005. Online "http://www.nemlar.org/Publications/Nemlar-report-ind-needs_web.pdf".
- [22]. Marques, N. C. & Gabriel, P. L. "Using neural nets for Portuguese part-of-speech tagging", proceedings of the 5th international conference on the cognitive science of natural language processing, Dublin City University, Ireland, 1996.
- [23]. Perschke, A. J. & Forcada M. L. "Part-of-speech tagging with recurrent neural networks", proceedings of the International Joint Conference on Neural Networks, IJCNN- IEEE2001:1588-1592, 2001.
- [24]. Principe, J.C., Euliano, N.R. & Lefebvre, W.C. "Neural and adaptive systems, fundamentals through simulations", John Wiley & Sons, NY, 2000.
- [25]. Rababaa, M., Batiha, K., & Jabar H. Yousif, "Towards Information Extraction System Based Arabic Language", International Journal of Soft Computing, Vol.1, No.1, ISSN: 1816-9503 PP 67-70, 2006.
<http://medwelljournals.com/abstract/?doi=ijscmp.2006.67.70>
- [26]. Schmid, H. "Part-of-speech tagging with neural networks", proceedings of COLING-94, Kyoto, Japan, pp 172–176, 1994.
- [27]. Srinivas, M. & Patnaik, L. M. Genetic algorithms: a survey", *IEEE Computer* 27(6), pp17-26, 1994.
- [28]. Ueffing, N. Macherey, K. & Ney, H. 2003. Confidence measures for statistical machine translation. New Orleans: MTSummit.
- [29]. Weischedel, R., et al. "Coping with ambiguity and unknown words through probabilistic models", Computational Linguistics. 19(2), pp359-382, 1993.
- [30]. Ying Dai, Basabi Chakraborty and Minghui Shi. "Kansei Engineering and Soft Computing: Theory and Practice." IGI Global, 2011. 1-436. Web. 19 Mar. 2012. doi:10.4018/978-1-61692-797-4.
- [31]. Yousif, J. (2019). Hidden Markov Model tagger for applications based Arabic text: A review. Journal of Computation and Applied Sciences IJOCAAS, 7(1).
- [32]. Yousif, J. H. (2013). Natural language processing based soft computing techniques. International Journal of Computer Applications, 77(8).
- [33]. Yousif, J. H., & Sembok, T. (2010, April). Automatic Part of Speech Tagger Based Arabic Language. In First joint scientific symposium of the colleges of applied sciences in the sultanate of Oman. Technological Development: Challenges and Perspectives (pp. 12-13).
- [34]. Yousif, J., & Al-Risi, M. (2019). Part of Speech Tagger for Arabic Text Based Support Vector Machines: A Review. ICTACT Journal on Soft Computing: DOI, 10.

Figure 4: Average MSE with Standard Deviation Boundaries for 5 Runs of MLP

Figure 5: Average MSE with Standard Deviation Boundaries for 5 Runs of FRNN

Figure 6: Average MSE with Standard Deviation Boundaries for 5 Runs of SVM

Table4: The comparison results of proposed taggers & other taggers

	Schmid[17]	Pérez[15]	Ahmed[1]	Khoja[11]	Diab[8]	Jabar[12]	Jabar[13]	Jabar[14]
Method Used	NN	DT-RNN	NN	Rule base	SVM	NN	NN	NN
Tag-set type	English	English	English	Arabic	Arabic	Arabic	Arabic	Arabic
Corpus size*10⁵	45	0.465	0.015662	0.5	0.04519 sentences	0.5	0.5	0.5
Train Data percent	44.4%	100%	85%	100%	80%	10%	10%	10%
NO. of Tag size	48	19	48	131	19	131	131	131
Accuracy %	96.22	92	90.4	90	94.5	99	99	99

Figure7: the Compression of proposed Taggers with other researchers