Context-Sensitive Spell Checking Based on Field Association Terms Dictionaries

Mahmoud Rokaya, Abdallah Nahla and Sultan Aljahdali

College of Computer and Information Technology, Taif University, Saudi Arabia

Summary

A context-based spelling error is a spelling or typing error that turns an intended word into another word of language. Most of the method that tried to solve this problem were depended on the confusion sets. Confusion set are collection of words where each word in the confusion set is ambiguous with the other words in the same set. the machine learning and statistical methods depend on Fand external dictionaries based on the concepts of field association terms and the power link. This method joins between the advantages of statistical and machine learning method and the re-source based methods. The values of precision, recall and F indicates that the proposed algorithm can produce in average 90%, 70% and 78% respectively which means that the algorithm tends to produce a low percentage of false negative errors. The value of F indicates the strong of the algorithm. Finally an evaluation experiment was done for the WinSpell algorithm performance using the new automatic approach to produce confusion sets.

Key words:
Field Association Terms, Power Link, Context Spelling Checkers, WinSpell

1. Introduction

Since the first work by Glantz, 1957 [6] , a great deal of researches has taken place on the subject of spelling verification and correction [4].

An approximate word matching algorithm is required to identify errors in queries where little or no contextual information is available and using some measure of similarity, recommend words that are most similar to each misspelled word [15].

The problem of creating or developing algorithms for automatically catching and correcting spelling errors has become a primary challenge for researchers in the last few decades. Kukich, 1992, divided the spelling errors into three types, non-word errors, isolated word errors and real word errors [10]. In this paper we consider the real word errors. This is the class of real-word errors in which one correctly spelled word is substituted for another. Some of these errors result from simple typos (e.g., from + form, form + farm) or cognitive or phonetic lapses (e.g., there + their, ingenious + ingenuous); some are syntactic or grammatical mistakes, including the use of the wrong inflected form (e.g., arrives ~ arrive, was + were) or the wrong function word (e.g., for + of, his ~ her); others are semantic anomalies (e.g., in five minutes, lave a message); and still others are due to insertions or deletions of whole words (e.g., the system has been operating system for almost three years, at absolutely extra cost) or improper spacing, including both splits and run-ons (e.g., myself ~ myself, ad here - adhere). These errors all seem to require information from the surrounding context for both detection and correction. Contextual information would be helpful also for improving correction accuracy for detectable non-word errors. [10].

The methods which tried to solve this problem fall in two classes: the first class is those methods that based on human made lexical, the other class is those methods that based on statistics or machine language.

An example of the first class is the method of Hirst and Budanitsky, 2005. They presented a method for correcting real-word spelling errors by restoring lexical cohesion [8]. This method detects and corrects real word spelling errors by identifying tokens that are semantically unrelated to their context and are spelling variations of words that would be related to the context. Relatedness to context is determined by a measure of semantic distance initially proposed by [9].

An example of the second class is the method of Wilcox-O'Hearn et. Al., 2008 [16]. They presented a statistical method based on trigrams for correcting real-word spelling correction. In this method, they made a reconsideration of the trigram-based noisy-channel model of real-word spelling-error correction that was presented by Mays, and Mercer in 1991[17] which has never been adequately evaluated or compared with other methods. They analyzed the advantages and limitations of the method, and presented a new evaluation that enables a meaningful comparison with the Word Net-based method of Hirst and Budanitsky.

Typically, the machine learning and statistical approaches rely on pre-defined confusion sets, which are sets (usually pairs) of commonly confounded words, such as {their, there, they're} and {principle, principal}. The methods learn the characteristics of typical context for each member of the set and detect situations in which one member occurs in context that is more typical of another. Such methods, therefore, are inherently limited to a set of

Manuscript received May 5, 2012
Manuscript revised May 20, 2012

common, predefined errors, but such errors can include both content and function words [16].

By contrast, the resource-based methods are not limited in this way, and can potentially detect a confounding of any two words listed in the resource that are spelling variations of one another, but these methods can operate only on errors in which both the error and the intended word are content words [16].

Rokaya and Atlam, 2010, proposed the concept of power link. The power link algorithm was suggested to measure how tow terms tend to appear together in a given corps. If the value of the power link between two terms was high then the chance that one of the terms is substituted using the other term is low. This means that those two terms cannot be confused [13].

The Winnow approach of Golding and Roth (1999) uses a multiplicative weight update algorithm that achieves a good accuracy and handles a large number of features [3]. The method learns large set of features with the corresponding weight. The method performs better than Bayesian. The multiplicative weight update algorithm represents the members of a confusion set as clouds of simple nodes corresponding to context words and collocation features. Winnow requires confusion sets to be known in advance. [7]

In this work we will try to recover the limitation of predefined errors by presenting an algorithm which is capable of detecting the errors. This means that the algorithm will start by checking every token in a given document and it will determine the candidates that can replace this given token. The number of the alternative can by as much as the algorithm can guess. If the number of the alternative exceeds three the power algorithm will be used to decide what terms should be removed from the confusion set.

Arabic's rich morphology (word construction) and complex orthography (writing system) present unique challenges for automatic spell checking. [11]. So this study will begin by applying the algorithms to English to prove its validity then it can be extended to other language specially Arabic.

The remaining sections of this paper are organized as follows. Section 2 reviews the concept of the power link and its expected role in the real word errors. Section3 presents summary of the WinSpell method. Section4 provides the experiments and its results of the proposed approach

2. The Power Link

The term power link was proposed by Rokaya and Atlam, 2010, as a method of building a dynamic field association terms dictionary. Power link algorithm presented a new rules to improve the quality of filed association terms (FATs) dictionary in English [12] . The origin of this concept comes from the co-word analysis researches. Co-word analysis considers the dynamics of science as a result of actor strategies. Changes in the content of a subject area are the combined effect of a large number of individual strategies. This technique should allow us in principle to identity the actors and explain the global dynamic [5].

If any two terms t_1 and t_2 belongs to a document D we will say that there is a link between t_1 and t_2. The power of this link will be measured by the function $\mathrm{LT}(\mathrm{t}_1, \mathrm{t}_{12})$ where:

$$LT((t_1,t_2)) = \frac{|D| \times cr(t_1,t_2)}{average_{i,j}L(((t_1,t_2))} \qquad (1),$$

where |D| is the number of different terms in the document D, $cr(t_1,t_2)$ is the co-occurrence frequency of t_1 and t_2 in the document D and $average_{i,j}L(t_{1i},t_{2j})$ represents the average distance between any instants $t_{1i}\ and\ t_{2j}$ of the terms t_1 and t_2 in the document D. For more details see Rokaya and Atlam, 2010 [13].

To estimate the power like between two terms t_1 and t_2 over a given corps we define the function $LCORPS(t_1,t_2)$. This function can be defined as:

$$LCORPS(t_1,t_2) = average_{D\in corps}LT(t_1,t_2) \qquad (2)$$

This function states that the terms t_1 and t_2 will tend to appear nearer together if the value of this function reasonably high. To give a threshold many values were experimentally has been tried around the mean value for the power link. This means that the threshold is not unique and it is dynamic. In fact it is dependent on the given corps. For our experiments the mean value was 31.5 so we activated the algorithm with values around this mean to cover the interval (mean-STD, mean+STD), where STD is the standard deviation of the mean value.

Mangnes, 2005, suggested a way to process large amounts of raw data, with the use of an approximate search algorithm to help focusing on interesting areas of a digital media [2].

In what follows we explain the partionning algorithm which is used to produce the raw confusion sets and then to process these confusion sets to get the final confusion sets

Fig. 1 illustrates the partionning algorithm. Since all terms are real words then all terms are exist in the dictionary or they are a proper nouns. The proper nouns will be considered as a correct word if they are correctly spelled. For the other terms, not proper nouns, and belong to the dictionary, we find all terms belong to the dictionary that has a similarity greater than a given threshold. Then we add the linguistic variations to each term to its confusion set . This means that we got the same confusion sets if we used the same dictionary but training the algorithm using different corpses will result in different confusion sets and different processing.

The power link measure is used to partition the confusion sets that has many terms. Exactly, any confusion set that contains more than 3 terms should be checked for possible partitioning using the power link measure. Both the training corps and the refined confusion sets represents the input data for the context spelling checking algorithm.

Table 1 shows the calculations of confusion sets for samples raw confusion sets. The raw confusion sets are extracted using the principle of similarities mentioned earlier. Table2 shows the effect of applying the power link algorithm to the confusion sets in Table1. The power link algorithm succeeded to divide the big confusion sets into a smaller confusion sets. To proof the validity of using the power link algorithm we established our experiments evaluate the automatic generated confusion sets using precision, recall and F measures ,then we applied the WinSpell algorithm to the resulting confusion sets .

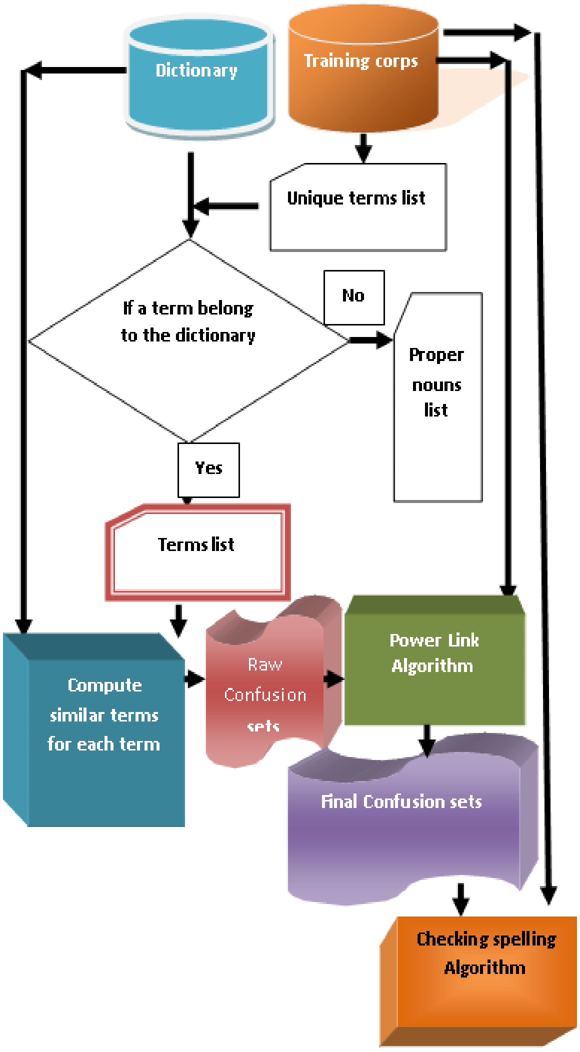

Fig. 1 Automatic construction of confusion sets

Table1: Some extracted raw confusion sets

RCS1	RCS2	RCS3	RCS4	RCS5	RCS6	RCS7
Are Area Aria Aura era	Gab Gad Gag Gal gap	Three Threw Throe There thru	Tag Tare Tags Take stage	Take Tike Tee Eke tyke	Lace Lice Lee Like lake	Alit Elite Alight Alike alive

Table 2: The resulting confusion sets after applying power link algorithm.

CS11	CS21	CS31	CS41	CS51	CS61	CS71
Are Area	Gab Gal gap	Three There	Tag Stage	Take Tike	Lace Lice	Alit Alight
CS12	CS22	CS32	CS42	CS52	CS62	CS72
Aria Aura era	Gad Gag	Threw Thru	Take Tare	Tee Eke tyke	Lee Like lake	Alike alive
CS13	CS23	CS33	CS43	CS53	CS63	CS73
		Throe				Elite

3. WinSpell

There are many methods for using a learning algorithm. Hidden Markov models are a powerful technique to model and classify temporal sequences, such as in speech and gesture recognition. However, defining these models is still an art: the designer has to establish by trial and error the number of hidden states, the relevant observations,.. ,etc. [18].

The WinSpell method has been among the most successful, and is thus the method we adopt here considering some vital modifications which gave some benefits regarding the automatic producing for the confusion sets and the performance of the algorithm. Briefly, we will review the algorithm and introduce our modifications.

WinSpell was introduced by Golding and Ruth, 1999. Winnow pursues the approach of learning the contextual characteristics of each word Wi individually. This learning can then be used to distinguish Wi word from any other word, as well as to perform a broad spectrum of other natural language tasks [3].

The approach developed was influenced by the Neuroidal system suggested by Valiant (1994) [19].The system consists of a very large number of items, in range of 105. The algorithm depend on handling a two levels of calculations. Winnow algorithm is originally designed for learning two-class (positive and negative class) problems, and can be extended to multiple-class problems [14]. The high level which combines the results of the lower levels. this means that the high level behave as a function in the lower level. Lower level consists of a number of classifiers. each classifier use the same method for calculations with different values. The high level use a cloud ensemble of the classifiers. Classifiers are combined using weighted majority algorithm. Fig. 2. shows the same example used by Golding and Roth (1999) to explain the idea. [3]

Let F be the set of active features; and for each active feature $f \in F$, let w_f be the weight on the arc connecting f to the classifier at hand. The winnow algorithm then returns a classification of 1 iff:

$$\sum_{f \in F} w_f > \theta$$

where θ is a threshold parameter.

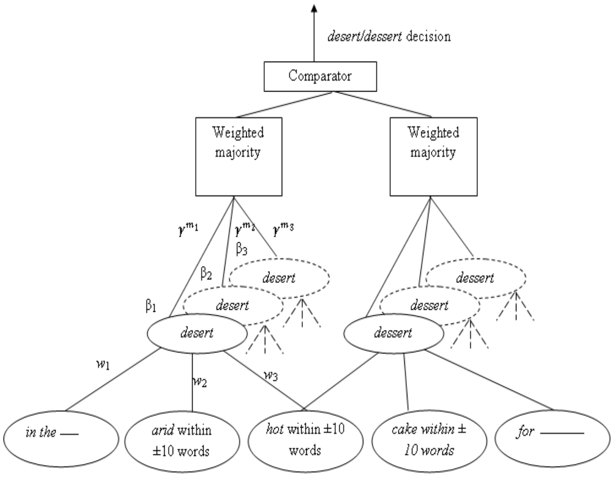

Fig. 2. Example of WinSpell network for {desert, dessert}

Initially, the classifier has no connection to any feature in the network. Through training it establishes appropriate connections, and learn weights for these connections. This means that the algorithm uses the policy of building connections on an as-needed basis results in a spare network with only those connections that have been demonstrated through training. The second step of training is to update the weights on the connections. the updating process happens only when an error is detected. There are two types of errors. If the classifier predicts 0 for a positive example then the weights are promoted:

$$\forall f \in F, w_f \leftarrow \alpha . w_f$$

where α > 1 is a promotion parameter. If the classifier predicts 1 for a negative example then the weights are demoted:

$$\forall f \in F, w_f \leftarrow \alpha . w_f$$

where β <1 is a demotion parameter.

Weighted majority means that a group of classifiers are trained using different distributions of training samples, and outputs of these classifiers are then combined in some manner to obtain the final classification rule [1].

The algorithm combines the results of different classifiers for the same word using the law of weighted majority :

$$\frac{\sum_j \gamma^{m_j} C_j}{\sum_j \gamma^{m_j}}$$

where C_j is the classification, either 1 or 0, returned by the jth classifier in the cloud. The weighting scheme assigns to the jth classifier a weight γ^{m_j}, $0 < \gamma < 1$ is a constant and mj is the total number of errors made by the classifier.

The next section explain the details of the evaluation process where we adopt the WinSpell algorithm with some minor modifications.

4. Evaluation.

To get a chance for positive comparison and fair results. We followed Hirst and Budanitsky (2005) [8] in using the 1987–89 Wall Street Journal corpus (approximately 30 million words), which we presume to be essentially free of errors. We reserved 500 articles (approximately 300,000 words) to create test data (see below).

To create a refined test sets, we automatically inserted real word errors in the reserved set of testing. Instead of using a fixed density distribution we used a varied density distribution. We used the properties of the power link to produce a balanced real errors e according to the following rule. Let ρ be the average of power link contained in a single document D, then the number of artificial errors added to this document is

$$e = \frac{N * \rho}{TN}$$

where, N is the number of unique words in D and TN is the total number of token in D. If the resulting number of errors e is greater than max ρ (max power link value in the document D) then, e is set to equal max ρ. Also if the resulting number of errors e is smaller than min ρ (min power link value in the document D) then, e is set to equal min ρ. This balanced inserting of real errors prevent to harm the natural power link distribution and guarantee an accepted density of real errors in each document. Note that the number of errors is proportional to the number of unique token in the document. This follows a simple rule that when many different words are written the probability for writing some errors is increased. Also putting the maximum and minimum of the power link as a boundaries for the number of errors guarantees that the inserted errors will not affect the distribution of the power link between terms in a single document. We define a spelling variation to be a single-character insertion, deletion, or replacement. [8]. We call this method, for insertion errors, FATI

In this evaluation two experiments are designed to test the new automatic generation of confusion sets and to test the performance of WinSpell under the new approach of inserting errors with respect to the predefined corps

To evaluate the automatic detecting of errors (finding the confusion sets). We prepared a set of manually detecting confusion sets. We combined these sets to get more greater confusion sets. Then we run the partionning algorithm with different values of the threshold and calculated the precision P, recall R and F measure values. Table 3 shows the results for different values of the threshold ϴ

The values of P, R and F show that the best performance is given near around ϴ=31.5 which presents the mean value of the power link. Also the precision values reflects that the performance of the algorithm is tend to include less number of false negative results than false positive results. The values of F insures the strong performance of the algorithm.

For the second experiments we followed the path of Wilcox-O'Hearn et. al. (2008) [16].

Table 3. Precision, Recall and F values for automatic producing for confusion sets

ϴ	***P***	***R***	***F***
17.9	0.639	0.483	0.550
21.8	0.665	0.749	0.705
25.7	0.779	0.716	0.746
29.6	0.928	0.687	0.789
31.5	0.963	0.696	0.808
33.4	0.812	0.729	0.768
35.4	0.760	0.720	0.740
37.3	0.644	0.686	0.664
39.2	0.521	0.547	0.534

They created three test sets, each containing 15,555 sentences, which varied according to which words were candidates for replacement and for substitution:

T20: Any word in the 20,000-word vocabulary of the trigram model could be replaced by a spelling variation from the same vocabulary; this replicates MDM's style of test set.

T62: Any word in the 62,000 most frequent words in the corpus could be replaced by a spelling variation from the same vocabulary; this reflects real typing errors much better than T20.

Mal: Any content word listed as a noun in Word-Net (but regardless of whether it was used as a noun in the text; there was no syntactic analysis) could be replaced by any spelling variation found in the lexicon of the ISPELL spelling checker; this replicates Hirst and Budanitsky's "malapropism" data [8].

Every confusion set was tested to classify it according to each of the three classes. For confusion sets that does not belong to any of these classes we placed it in a fourth class. We applied our approach to this class and used it as a parameter to test our approach independently. In the results, we called that class MFATC.

We applied WinSpell to each of the classes. First set of experiments were applied to test the WinSpell. Table4 indicates that the performance of WinSpell is lower than the performance of Wilcox-O'Hearn et. al. (2008) [16].

The results in Table4 for precision and recall insures the poor performance for the WinSpell. There is no significance difference in the performance of the algorithm among different test groups. these results open the door for the question: Is there some modification that can be done to improve the performance of WinSpell? The answer is yes. Considering the new researches for the weighted majority rules and instead of imposing an external rules that define the thresholds in pruning and combining the voting classifiers we can adopt values that depend on the co-occurrence information of terms in a given corps.

Table 4: results of applying WinSpell without modification

	Detection			***Correction***		
α	***P***	***R***	***F***	***P***	***R***	***F***
Test set T20						
0.9	0.334	0.647	0.441	0.327	0.618	0.428
0.99	0.474	0.668	0.555	0.467	0.547	0.504
0.995	0.546	0.636	0.588	0.539	0.616	0.575
0.999	0.594	0.658	0.624	0.690	0.543	0.608
FATI	0.529	0.559	0.544	0.607	0.440	0.510
Test set T62						
0.9	0.235	0.437	0.306	0.229	0.419	0.296
0.99	0.347	0.478	0.402	0.341	0.366	0.353
0.995	0.423	0.460	0.441	0.417	0.350	0.381
0.999	0.593	0.400	0.478	0.590	0.395	0.473
FATI	0.599	0.396	0.477	0.667	0.438	0.529
Test set Mal						
0.9	0.145	0.367	0.208	0.140	0.352	0.200
0.99	0.306	0.320	0.313	0.299	0.310	0.304
0.995	0.371	0.304	0.334	0.365	0.296	0.327
0.999	0.446	0.261	0.329	0.443	0.257	0.325
FATI	0.379	0.313	0.343	0.421	0.239	0.305
Test set MFATC						
0.9	0.112	0.496	0.183	0.105	0.471	0.172
0.99	0.298	0.436	0.354	0.29	0.419	0.343
0.995	0.359	0.41	0.383	0.353	0.397	0.374
0.999	0.52	0.344	0.414	0.516	0.336	0.407
FATI	0.588	0.378	0.460	0.573	0.380	0.457

5. Conclusion

In this work we proposed a method of automatic producing of confusion sets (errors) for a given dictionary of terms and a corresponding corps. The results shows that this method retrieve 96% of confusion sets and gives the correct divisions of these errors. The results insures that the algorithm can retrieve confusion sets with false positive errors less than its ability to retrieve false negative errors. Also we evaluated the performance of WinSpell algorithm based on the automatic constructed confusion sets. The results prove that the WinSpell algorithm reflected a power performance and it needs some modifications to improve the WinSpell. algorithm reflected a power performance and it needs some modifications to improve the WinSpell.

Acknowledgments

The authors gratefully acknowledge the Taif University for the financial support (project No: 1-432-1204)

References

[1] Aliasgar Gangardiwala and Robi Polikar, Dynamically Weighted Majority Voting for Incremental Learning and Comparison of Three Boosting Based Approaches,Proceedings of International Joint Conference on Neural Networks, Montreal, Canada, July 31 - August 4, 2005, 1131-1136

[2] Bjarne Mangnes, The use of Levenshtein distance in computer forensics, Master thesis, Gjøvik University College, 2005.

[3] Andrew R. Golding and Dan Roth, A Winnow-Based Approach to Context-Sensitive Spelling Correction, Machine Learning 34, 107–130 (1999)

[4] Boubaker Meddeb Hamrouni, LOGIC COMPRESSION OF DICTIONARIES FOR MULTILINGUAL SPELLING CHECKERS, August 1994 COLING '94: Proceedings of the 15th conference on Computational linguistics - Volume 1, 293-296

[5] Callon, M., Courtid, J. and Ladle, F. (1991) 'Co-word analysis as a tool for describing the network of interactions between basic and technological research: the case of polymer chemistry', Science Metrics, Vol. 22, No. 1, pp.155–205.

[6] Glantz , (1957)On the recognition of infornultion with a digital computer, J. ACM, Vol. 4, No. 2, 178-188.

[7] H. Al-Mubaid and K. Truemper, "Learning to Find Context-Based Spelling Errors", in Data Mining and Knowledge Discovery Approaches Based on Rule Induction Techniques, E. Triantaphyllou and G. Felici (Editors). Massive Computing Series, Springer, Heidelberg, Germany, pp. 597-628, 2006

[8] Hirst, Graeme and Budanitsky, Alexander (2005). Correcting real-word spelling errors by restoring lexical cohesion. Natural Language Engineering, 11(1), 87–111.

[9] Jay J. Jiang and David W. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. Proceedings of 10th Interna-tional Conference on Research In Computational Linguistics, 1997.URL http://www.citebase.org/abstract?id=oai:arXiv.org:cmp-lg/9709008

[10] Karen Kukich: Techniques for Automatically Correcting Words in Text. ACM Comput. Surv. 24(4): 377-439 (1992)

[11] Khaled Fouad Shaalan, Amin Allam & Abdallah Gomaah, Towards Automatic Spell Checking for Arabic, Conference on Language Engineering, Cairo, pp 240-247, 2003

[12] Mahmoud Rokaya and Abdallah Nahla, Building a Multi-lingual Field Association Terms Dictionary, International Journal of Computer Science and Network Security, Vol 11, No. 3, March, 2011

[13] Mahmoud Rokaya and Atlam, E-S. (2010) 'Building of field association terms based on links', Int. J. Computer Applications in Technology, Vol. 38, No. 4, pp.298–305.

[14] Surapant Meknavin, Combining Trigram and Winnow in Thai OCR Error Correction, COLING '98 Proceedings of the 17th international conference on Computational linguistics - Volume 2, 836-842

[15] Victoria J. Hodge and Jim Austin, A Comparison of Standard Spell Checking Algorithms and A Novel Binary Neural Approach, IEEE transactions on knowledge and data engineering, vol. 15, no. 5, September/October 2003. 1073-1081

[16] Wilcox-O'Hearn, L. Amber; Hirst, Graeme; and Budanitsky, Alexander (2008). Real-word spelling correction with trigrams: A reconsideration of the Mays, Damerau, and Mercer model. Proceedings, 9th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2008), Haifa, 605–616.

[17] Eric Mays, Fred J. Damerau, and Robert L. Mercer. 1991. Context based spelling correction. Information Processing and Management, 23(5):517-522.

[18] Miriam Mart´ınez and L. Enrique Sucar, Learning Dynamic Naive Bayesian Classifiers, Proceedings of the Twenty-First International FLAIRS Conference (2008), 655-659

[19] Valiant, L.G. (1994). Circuits of the mind. Oxford University Press.

Mahmoud Rokaya received B.Sc. and M. Sc. Degrees in Mathematics from, Faculty of Science, Tanta University, Egypt, in 1997 and 2003 respectively, and the Ph.D. degree in information science and Intelligent systems from University of Tokushima, Japan, in 2009. He is currently an assistant professor in Dept. of Statistical and Computer science, Tanta University, Egypt. Mahmoud is a member in Egyptian Mathematical Association (EMA). His research interests include information retrieval, natural language processing and document processing.

Abdallah Nahla received B.Sc. and M. Sc. Degrees in Mathematics from, Faculty of Science, Tanta University, Egypt, in 1997 and 2003 respectively, and the Ph.D. degree in nuclear reactors from University of Tanta, Egypt, in 2006. He is currently an assistant professor in Dept. of Mathematics, Taif University, Saudi. Abdallah is a member in

Egyptian Mathematical Association (EMA). His research interests include partial differential equations, nuclear reactors and information systems.

Sultan Aljahdali received B.Sc. and M. Sc. Degree in Computer Science from, Winona State University, Winona, Minnesota, U.S.A, in 1992, and 1996 respectively, and the Ph.D. degree in Information Technology from Minnesota State University, Mankato, Minnesota, U.S.A, in 2003. He is currently an associate professor in Dept. of Computer Science, Taif University. Also Dr. Sultan is the dean of college of Computer & Information Technology, Taif University Saudi Arabia.. His research interests include Performance Prediction in Software Engineering, Software Modeling and Cost Estimation, Performance Modeling.