The International Arab Journal of Information Technology, Vol. 4, No. 3, July 2007

A Learning-Classification Based Appro Word Prediction

Hisham Al-Mubaid
Computer Science Department, University of Houston-Clear Lake, U

Abstract: *Word prediction is an important NLP problem in which we want to predict the correct w Word completion utilities, predictive text entry systems, writing aids, and language translation are prediction applications. This paper presents a new word prediction approach based on context features The proposed method casts the problem as a learning-classification task by training word discriminating features selected by various feature selection techniques. The contribution of this work presenting this problem, and the unique combination of a top performer in machine learning, svn selection techniques MI, X^2, and more. The method is implemented and evaluated using several dat results show clearly that the method is effective in predicting the correct words by utilizing sma achieved impressive results, compared with similar work; the accuracy in some experiments ap predictions.*

Keywords: *Word prediction, word completion, machine learning, natural language processing.*

Received January 7, 2006; accepted June 6, 2006

---

1. Introduction

Word Prediction (WP) is an important Natural Language Processing (NLP) task in which we want to predict (*determine*) the correct word in a given context. Word prediction task can be employed in many applications, for example, predictive text entry systems, word completion utilities, and writing aids [9, 13]. Statistical and similarity based approaches have done quite well in tackling this problem just like other similar problems such as *word sense disambiguation* [4, 12, 21, 22, 23]. A common approach to handle such disambiguation-like problems is to train and apply word *bigram* or *n-gram* models.

This paper presents an effective method for word prediction using machine learning and new feature extraction and selection techniques. We use feature selection techniques adapted from Mutual Information (MI) and Chi-square (X^2). These feature extraction and selection techniques, *MI* and X^2, have been used successfully in Information Retrieval (IR) and Text Categorization (TC) [10, 11, 26]. Thus, the *WP* problem here is casted as a word classification task in which multiple candidate words are classified to determine the most correct one in the given context. For example, in this word prediction instance:

$$[w_n \ldots w_3\, w_2\, w_1 \text{ -?- }]$$

we wish to predict and determine the word that follows the sequence … $w_3\, w_2\, w_1$ (i. e., the word in place of the "-?-").

The proposed method has a un the representations of words in a g

1. For a given occurrence o representation of *w* invol occurrence of certain word fe the training corpus using ne techniques adapted from *MI* an
2. The encoding of (1) is used in train word classifiers using the
3. The word classifiers of (2) a word predictors in a new w correct word given its context. of this method is that it perfo very small contexts (only prece

The method has been implem extensively; the experiments and this paper. The results clearly method is effective in predicti utilizing very small contexts. T accuracy approaching *91%* in so outperforming most of the publi task.

The rest of the paper is o Section 2 presents a brief overvie The proposed methods includi learning, and prediction are ex Section 4 describes the bas evaluation process and exper discussed in section 5. Finally, s conclusion.

2. Related Work

A number of methods and systems have been proposed for word prediction in the past few decades. These methods can be classified as statistical methods that are based on statistical (and probabilistic) language models; and syntactic methods in which syntactic information is extracted and exploited in word prediction task. In [9], Fazly presents a comprehensive review of prior related work in word prediction. Fazly also presents a collection of experiments on word prediction applied to word completion utilities. The implemented and evaluated algorithms [9] were based on word unigrams and bigrams, and based on syntactic features like POS tags in the syntactic predictors, and combination. The training and testing are done on texts taken from British National Corpus (BNC). Roughly speaking, tags-and-words predictors achieved the best overall performance with *hit rate* approaching *37%*, and keystroke savings around *53%* —*hit rate* is defined to be the percentage of the times that the correct word appears in the prediction list. Among the other related interesting work is the approach presented in [7]. That approach attempts to learn the contexts in which a word tends to appear, using expressive and rich set of features. The features are introduced in a language as information sources. It also attempts to augment local context information by global sentence information. The evaluation of the method in this paper is very similar to that presented in [7].

One of the related problems to word prediction is the context-sensitive spelling error correction, or *malapropisms* [2, 14]. In this problem, the misspelled variant of the original word is a correct word and belongs to the language [2, 14, 15]. For example, the misspelling of the word *quite* as *quiet* is a context-sensitive spelling error. Since *quiet* is a valid word in *English*, the traditional spell-checkers will not discover this spelling error. Thus, the function of the context-sensitive spelling correction is to choose, for an instance for a word in text (e. g., *quite)*, its correct spelling from its confusion set (e. g., *quite, quiet*). It is worth mentioning at this point that *word prediction* can be harder than *context-sensitive spelling problem* such that, in the latter problem the size of the given context is double the size of the given context in word prediction. That is, in word prediction, only the preceding words are available as context to the prediction task, whereas in the context-sensitive spelling correction task, the words before and after the target word are available as a context. Of course the context of prediction or classification task is critically an important resource for such a task.

3. The Proposed Method

The proposed method is based on representing each word as a feature vector, then using machine learning to train word classifiers during th
word classifiers are then emplo
phase, to determine from a conf
word in a given context. Thus, tl
word classification task. For exam
set be {*weak, week*} then when a
'*w*' the word prediction task t
determine whether the user wan
'*week*'.

In a given context (e. g., [w_n,
want to predict the word w_x, such t
word to be predicted (e. g., {w
given along with the confusion s
is the set of the alternative (*can*
context, e. g., {w_x, w_y}. We want
which of the two *candidate* word
in this context. In word completi
prediction task can start after typ
the target word, so that, the pr
limited to alternative words that s
letter. In this research, we fol
researchers and assume that th
predetermined [7, 8, 14, 15].
contains two or more of the most
the language. For example, MS
list of confusion sets, called
words, for grammar checking. S
Table 1, can be used as a basis fo

Table 1. A part from the commonly co
Word.

Commonly Confused '
Abut-About, Adept-Adapt, Adepts-Adopts, Aid-Aide, Ail-Ale, Alters-Altars, Assess-A Bear, Beet-Beat, Bettor-Better, Border- Bridal-Bridle, Broach-Brooch, …
…
…
Theirs-Their's, Tide-Tied, Undo-Undue, Up Vein-Vain, Who's-Whose, Wile-While, Wit Yolk-Yoke, You're-Your

Examples of confusion sets u
include: {*quite-quiet, peace-piece*
begin, than-then, raise-rise, site-s
we can summarize the problem, a

Let c = {w_1, w_2, ..., w_n} b
prediction task.

where n is an integer numbe
size of context window (*in this re*
values of 3, 5, or 10).

The words w_1, w_2, ..., w_n are t
immediately before the word to be
= {w_x, w_y} be the confusion set
proposed method relies on mach
word classifiers to classify (predic
the predicted correct word in tha
in the confusion set is represente
the feature vector that is compos
data. One of the contributions o

way we extract and compute the features from the training data. We describe next the feature extraction process and then we talk about the learning and the prediction steps.

3.1. Feature Selection and Extraction

Let a training text T be given. We extract from T all the occurrences of the confusion set words w_x and w_y. Each occurrence is extracted along with its context (preceding n words) to make one *training* example of the form [w_n ... w_3 w_2 w_1 $\underline{w_x}$] or [w_n ...w_3 w_2 w_1 $\underline{w_y}$]. Thus, we have now two sets of training examples; the training examples of w_x and the training examples of w_y both extracted from T. We convert each example into a feature vector as follows. The given context words are used as features in some of the related work [14, 22, 23]. In this research, however, we do not use word features directly from the contexts; instead we select, as features, only certain words with high "*discriminating*" capabilities between the two confused words (w_x and w_y). These features are used to represent each example in the training and prediction. We use the confusion words occurrences extracted from the training text T as labeled training examples. Feature selection is a key issue in the effectiveness and efficiency of the learning and classification performance of such methods as the one presented here.

Before delving into the details of feature selection, let us mention that there has been a lot of research devoted to feature selection in machine learning and data mining, particularly in *text categorization* research, see for example [10, 11, 26]. Assume that we have two classes C_1 and C_2 of labeled examples extracted from the training text T. Let C_1 contains examples of w_x and their contexts, and C_2 includes examples of w_y with their contexts. We extract all the context words $W = \{w_1, w_2, ..., w_m\}$ from the sets C_1 and C_2. Now, each such context word $w_i \in W$ may occur in contexts from C_1 or C_2 or both with different frequency distributions. Now, if a context word $w_i \in W$ appears in a context of a prediction example, we would like to be able to determine to what extent the existence of w_i suggests that this example belongs to C_1 or C_2. Thus, we select those words w_i from W which are highly associated with either C_1 or C_2 (*the highly discriminating words*) as features. We utilize feature selection techniques like *MI* and X^2 [11, 26] to select the highly discriminating context words from W. MI and X^2 were used effectively for feature selection in text categorization and information retrieval [10, 11, 26] but never been utilized for language prediction or classification problems. In the rest of this section, we explain how *MI* and X^2 are applied to determine which context words from W will be selected as features.

Let us first define the notions of *a, b, c,* and *d* as follows. From the training examples, we calculate four numeric values a, b, c, and d fo
$w_i \in W$ as follows:

a = Number of occurrences of w
b = Number of occurrences of w
c = Number of examples of C_1 t
d = Number of examples of C_2 t

Then, MI is defined as:

$$MI = \frac{N*a}{(a+b)*(a+c)}$$

Where N is the total number of ex
Chi-square (X^2) is computed as:

$$X^2 = \frac{N*(ad-cb)^2}{(a+c)*(b+d)*(a+b)*}$$

Again, N is the total number of ex

Illustrating the proposed WP m
when using the *MI* technique for
calculate the *MI* value for each
choose the k top $w_i \in W$ words
values as features. In our experi
values of 10, 20, and 30. For exa
each training example is represen
entries, such that, the first entry
with the highest *MI* value, the se
the word with the second highest
Then for a given training examp
entry is set to 1 if the corresp
occurs/appears in that training e
otherwise. Thus, if we want to
discriminating words as feature
example, then feature vector size
the following example, let W = {
set of all context words. We c
$w_i \in W$ and sort the words W a
values in descending order as in T

Table 2. Words $w_i \in W$ with the

Context Words w_i	M
Person	1.
Nice	1.
Found	1.
Still	1.
Place	1.
Generate	1.
Went	1.
Clear	1.
Deliver	1.
Small	1.
…	.

Table 2 shows the top 10 cont
highest 10 MI values. These 10 v
compose the feature vectors for tr
examples. For example, the follov

0	*1*	*1*	*0*	*0*	*0*	*1*			

represents an example containing the 2nd, 3rd and 7th feature words (*viz., nice, found,* and *went)* in the given context. Additionally, if the window size is 5, then that example may look like:

went ___ nice ___ found $< w_x$ *or* $w_y >$

That is, three of the 10 feature words are occurring within the preceding 5 words of the word to be predicted. In this case, window size is 5 and the vector size is 10. For example, the word '*nice'*, occurred as third preceding word in the context but it is translated to a '1' in the seventh entry of the feature vector.

Let us look into the MI feature selection technique in little more detail. The objective of MI is to select from two classes C_1 and C_2 of examples the most discriminating features (*words*). A good such feature is the one that is highly associated with C_1 but not with C_2 or vice versa. MI uses the co-occurrence counts *a*, *b*, *c*, and *d* with equation (1) to compute MI value for each feature, such that the feature with highest MI value will be the best in discriminating C_1 from C_2. The MI's formula gives most weight to *a* (the numerator in equation (1)) which represent the association between the word/ feature and class C_1. We would like to update this formula by multiplying MI by the difference ($a – b$) between *a* and *b*. Recall that, for a given word w_i, the value *b* represents the association between w_i and class C_2 (how many times w_i occurs in C_2). In this, we subtract from *a* the number of times the word is associated with C_2. That is, if a word w_i is associated *q* times with C_1 and *q* times with C2 then the formula yields zero, which is what we want, since in this case, the feature w_i is not really a discriminating feature. Thus, we applied the formula:

$$MI\_1 = MI * (a - b) \tag{3}$$

for feature selection. Furthermore, to give more weight to *a*, we also applied the formula:

$$MI\_2 = MI * a * (a - b) \tag{4}$$

Notice that equations (3) and (4) can also be written as:

$$MI\_1 = \frac{N*a}{(a+b)*(a+c)} * (a - b)$$

$$MI\_2 = \frac{N*a}{(a+b)*(a+c)} * a * (a - b)$$

respectively.

We found out after extensive experimentation, with multiple datasets, that using *MI_2* for feature selection gives, in most cases, better results than *MI* and *MI_1*, see Table 3. The results in Table 3 demonstrate clearly that our proposed feature selection technique *MI_2* which is adapted from *MI* outperforms *MI* across the three confusion sets using *Reuters* dataset. These experiments as shown in Table 3 are done on more than 3,000 prediction instances (*Table 6 gives numbers of testing instances in Reuters and other datasets*).

Thus, in our experiments we use
of *MI* or *MI_1*) and X^2 for feature

Table 3. Accuracy results of four featur
three confusion sets using *Reuters* datase

Confusion Sets	MI	MI*A		
Conf. set 1	72.77	79.71		
Conf. set 2	86.32	88.87		
Conf. set 3	92.77	94.79		

3.2. Learning and Prediction

Thus, from the training text
vectors using the top words select
Then, we use the well-establishe
Support Vector Machines (SV
classifiers with the training v
inductive learning techniqu
classification. A significant elab
and empirical justification has b
literature to support SVM [3, 6].
extensively applied in various
remarkable results.

For example, in text categ
investigated extensively and pro
best learning algorithms [6, 10,
method, for a given confusio
construct one feature vector for
instance in the training text. Thus
the training examples, and we
classes, one for w_x vectors and on
SVM trains on these two clas
classifier (*model*). Thus, we co
classifier for each confusion set.
is then used in the prediction pha
in the given context. Of cours
process, we construct a feature v
as in the training process. We us
our experiments as most of
implementation of SVM we use
light, available at: http://svmlight
default parameters.

4. The Baseline Method: *N*

We applied *Naïve Bayes* (N. Bay
task to compare with our meth
Bayes for *WP*, we followed the
assuming the probabilistic mo
examples [8]. *Naïve Bayes* wa
disambiguation-like NLP proble
sense disambiguation [4, 12, 21,
introduce *Naïve Bayes* here
experimental settings with it, for
refer to [12, 17]. Let $W = \{w$
context. Let further $C = \{c_1, c_2, ...$
set that contains the alternative (
the prediction task. The decisi
Bayes is as follows:

$$c^* = \underset{k}{\operatorname{argmax}}\ P\,(c_k|W) \ = \ \underset{k}{\operatorname{argmax}}\ (P(c_k)\ .\prod_{i=1}^{n} P\,(w_i\,|\,c_k)) \quad (5)$$

Such that $P\ (c_k\ |W)$ is the conditional probability of the confusion set word c_k appears in the context W. This decision rule selects $c^* \in C$ as the predicted word in the given context W. The probabilities P (c_k) and P ($w_i|c_k$) are computed from the training text T. Notice here that *Naïve Bayes* assumes that the context words $w_1, w_2, \ldots, w_n$ are conditionally independent. There is one issue with the *Naïve Bayes* is that the probability P ($w_i|c_k$) may, very well, be a very small value or zero, so we use a smoothing technique to avoid this problem. There are a number of smoothing techniques proposed in the literature, for example, *add-1*, *Ng's smoothing*, and *Kneser-Ney and Katz smoothing*. For more details on smoothing see [5, 14]. Chen *et al.* (1998) [5] presents a comprehensive review about the smoothing techniques.

Formatted: Font: 7 pt, Complex Script Font: 7 pt

5. Evaluation and Experimental Results

In this section, we describe the datasets used in experiments and the experimental settings, then we discuss the results.

5.1. Datasets

We used four different text datasets to evaluate our method. The details of the datasets are in Table 4. We select the testing text size to be little less than the training text size as the case in the actual prediction. The testing text size is not important and will not affect the performance as we only utilize the preceding 3 words for each prediction case. The datasets are as follows:

- The ACL dataset were obtained from Linguistic Data Consortium (LDC) (www.ldc.upenn.edu) and include news stories 1987-1991 taken from the Wall Street Journal (WSJ).
- The Reuters is taken from the Reuters-21578 benchmark dataset. Reuters-21578contains 21578 news articles from the Reuters newswire [24].
- The BioMed text is a corpus of biomedical articles taken from Medline [18]. The Medline database is considered to be the largest and most comprehensive data resource in bioinformatics. We use this text to evaluate the performance of our method on specialized texts.
- The 10-K dataset contains financial text of 10-K filings of US corporate, taken from U.S. Securities and Exchanges Commissions (SEC) at (www.sec.gov). 10-K filing is an annual financial and transactional report required by SEC from all public companies, and it comprehensive information on of a public company. At SEC of around 10,000 public comp years are available (and tota 50,000 filings. The size of around 30 GB.). This dataset i text (financial text) used to eva

Table 4. Details of the four datasets

Dataset (Source)	T T	
Reuters (Reuters-21578)		
ACL (LDC www.ldc.upenn.edu)		
Biomed Text (*Medline*)		
10-K (SEC at www.sec.gov)		

Table 5. The three confusion sets use

Confusion Set 1	Accept-except, aff country-county, …
Confusion Set 2	Site-sight, than-the rise, …
Confusion Set 3	Advice-advise, wea loose, …

5.2. Confusion Sets

We used three confusion sets shown in Table 5. These c commonly used in word pred sensitive spelling research; see [2,

5.3. Evaluation and Discussio

Several experiments have been c the method. We used *MI*, *MI_2* selection, and *SVM* for learning ar used the *N. Bayes* algorithm compare our results. For cor preceding 3, 5, or 10 words. We of size 3, using only preceding 3 best performance. Furthermore, how many features to include ir For that, we tried 10, 20, and 3 that the best performance resu features (i. e., using the top 2 highest 20 *MI_2*, or X^2). Thus, here are generated using the (context size = 3) and the top 2(We initially tested our metho *Reuters*, *ACL*, and *BioMed* (Tab confusion sets (Table 5). The re Table 6 when using *MI_2* for fea Table 7 when the X^2 feature sel used. With a total of 19,438 wor were tested in each experiment (notice that *MI_2* (Table 6) pro accuracy than X^2 (Table 7).

Moreover, to compare our method against the baseline method we ran all the testing prediction instances on the *Bayesian* method and the results are in Table 8. The Bayesian method produced slightly better accuracy than MI_2 only in the *Reuters* dataset, but with the other two datasets, both MI_2 and X^2 outperform Bayesian significantly (Table 8). Furthermore, the micro-average accuracy on the three datasets demonstrates that *MI_2* and X^2 outperform *Bayesian* (Table 8). Finally, since the *10-K* dataset is very specialized dataset and is not as commonly used in NLP research as the other datasets, we tested our method on it in a separate experiment using *MI_2* and X^2 with the three confusion sets, and the results are in Table 9. In this experiment too, *MI_2* with 91.42% accuracy outperforms X^2 with 87.09% accuracy. This experiment also proves that our method can achieve impressive accuracies exceeding 91% correct predictions (Table 9). Overall, our method of learning-classification-based word prediction is capable of achieving accuracy in the range of 87% – 88% correct predictions using only the three preceding words as context, which emphasizes the robustness of the feature selection techniques and the learning method. Furthermore, the experimental results proved that the method can achieve really high accuracies; for example, the method produced accuracy of ~90% using confusion set 2 and *Reuter* (Table 7), and the average accuracies on *BioMed* and *Reuters* are approaching ~89% and ~90%, respectively (Table 6). In addition, the method achieved accuracy of 95.2% on *Reuters* using confusion set 3 (Table 6) and 93.1% on the *BioMed* dataset using confusion set 3 (Table 7).

Table 6. Accuracy results with the 3 datasets and 3 confusion sets using *MI_2* for feature selection, preceding 3 words for contexts, and top 20 features.

Dataset	Confusion Set 1		Confusion Set 2		Confusion Set 3		Average Accuracy
	No. of Tested Instances	Accuracy	No. of Tested Instances	Accuracy	No. of Tested Instances	Accuracy	
Reuters	615	81.46	1481	89.80	941	95.21	89.79
ACL	2658	86.68	3149	83.39	2369	87.08	85.53
BioMed	2725	86.93	4313	88.73	1187	93.09	88.76
Total	5998		8943		4497		

Table 7. Accuracy results with the 3 datasets and 3 confusion sets using X2 for feature selection, preceding 3 words for contexts, and top 20 features.

Dataset	Confusion set 1		Confusion set 2		Confusion set 3		Average Accuracy
	No. of Tested Instances	Accuracy	No. of Tested Instances	Accuracy	No. of Tested Instances	Accuracy	
Reuters	615	81.46	1481	89.80	941	86.96	87.23
ACL	2658	85.94	3149	82.85	2369	87.21	85.12
BioMed	2725	85.13	4313	87.22	1187	93.09	87.37
Total	5998		8943		7853		

6. Contribution and Conclusion

We presented a learning-classification based method for word prediction. The method uses very small context (*the preceding three words*) to predict the following word in that context v
The method was evaluated extens
with the *Bayesian* algorithm a
experimental results showed tha
achieve impressive accuracy in pe
predictions, which validates i
contribution of this work can be
new aspects: Casting the *wp* t
classification task by using mach
word predictors using highly dis
selected by various techniques. Th
also includes a new feature select
adapted from *MI* and outperform
experiments. Furthermore, the uni
one of the top performers in mac
with feature selection techniques
are used in TC and IR, makes a
into *WP*. These aspects can co
other similar NLP problems as n
this paper.

Table 8. Average accuracy on each m
accuracy here is the average of testing on

Dataset	No. of Tested Instances	A		
		N.Bayes		
Reuters	3037	90.67		
ACL	8176	80.12		
BioMed	8225	81.28		
Total	19,438			
Micro. Avg		82.26		

Table 9. Accuracy results for th

Dataset	No. of Tested Instances	A	
		*MI_*	
10-K	2,610	91.4	

Word prediction is a very im
many significant applications.
prediction system can benefit user
text entry rates, and minin
typographical errors and misspell
been observed by the developers
word processor *OpenOffice* [2
along with standard word proce
completion (*www.openoffice.org*)
directions of this research, we wo
new aspects to further improve the
For example, we will investigate i
size without affecting the compu
the method. Also, we plan to exp
involving positional information
features in the learning process.

Acknowledgements

This work was supported by the Institute for Space Systems Operations (ISSO), February 2005.

References

[1] Al-Mubaid H., "Context-Based Word Prediction and Classification," *in Proceedings of the 18th International Conference on Computers and their Applications (CATA'2003)*, Hawaii, USA, pp. 384-388, March 2003.

[2] Al-Mubaid H. and Truemper K., "Learning to Find Context-Based Spelling Errors," in Triantaphyllou E. and Felici G. (Eds), *Data Mining and Knowledge Discovery Approaches Based on Rule Induction Techniques*, Kluwer Academic Publishers, 2006.

[3] Bose B. E. R., Guyon I., and Vapnik V., "A Training Algorithm for Optimal Margin Classifiers," *in Proceedings of COLT*, USA, pp. 144-152, 1992.

[4] Bruce R. and Wiebe J., "Word-Sense Disambiguation Using Decomposable Models," *in Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics (ACL'94)*, USA, pp. 139-145, 1994.

[5] Chen S. and Goodman J., "An Empirical Study of Smoothing Techniques for Language Modeling," *Technical Report TR-10-98*, Centre for Research in Computing Technology, Harvard University, Cambridge, Massachusetts, 1998.

[6] Dumais S. T., Platt J., Heckerman D., and Sahami M., "Inductive Learning Algorithms and Representations for Text Categorization," *in Proceedings of the 7th International Conference on Information and Knowledge Management*, USA, pp. 148-155, 1998.

[7] Even-Zohar Y. and Roth D., "A Classification Approach to Word Prediction," *in Proceedings of the NAACL'00*, USA, pp. 124-131, May 2000.

[8] Even-Zohar Y., Roth D., and Zelenko D., "Word Prediction and Clustering," *The Bar-Ilan Symposium on the Foundations of Artificial Intelligence*, June 1999.

[9] Fazly A., "The Use of Syntax in Word Completion Utilities," *Master Thesis*, University of Toronto, Canada, 2002.

[10] Forman G., "An Extensive Empirical Study of Feature Selection Metrics for Text Classification," *JMLR*, vol. 3, no. 1, pp. 1289-1305, 2003.

[11] Galavotti L., Sebastiani F., and Simi M., "Experiments on the Use of Feature Selection and Negative Evidence in Automated Text Categorization," *in Proce European Conference Advanced Technology fo (ECDL'00)*, Portugal, pp. 59

[12] Gale W. A., Church K. W., Method for Disambiguatin Large Corpus," *Computers* vol. 26, no. 1, pp. 415-439,

[13] Garay-Vitoria N. and G "Intelligent Word-Predictio Input Rate," *in Procee International Conference Interfaces*, pp. 241-244, Jan

[14] Ginter F., Boberg J., Jarvi T., "New Techniques for Natural Language and T Biological Text," *JMLR*, v 621, 2004.

[15] Golding A. R. and Roth D. Approach to Context- Correction, in Machine Lea *on Natural Language Lear* pp. 107-130, 1999.

[16] Joachims T., "Text Catego Vector Machines: Learning Features," *in Proceedings Conference on Machine Le* 1998.

[17] Manning C. D. and Sch¨utz *Statistical Natural Langua* Press, Cambridge, Massach

[18] Medline, Accessed Usin Interface, available at: nih.gov/entrez/query.fcgi , 2

[19] MS WORD: Microsoft Cor http://www.microsoft.com/

[20] OpenOffice, *A Multiplatfo Office Suite and Open-Sou* and Sponsored by Sun Mi at: http://www.openoffice.o

[21] Pedersen T., Bruce R., and Model Selection fo Disambiguation," *in Pr Conference on Applied Processing (ANLP'97)*, W 388-395, 1997.

[22] Pedersen T. and Bruce R Word Sense Disambiguatio *the 15th National Confe Intelligence (AAAI'98)*, Mad

[23] Pedersen T., "Search Tec Probabilistic Models Disambiguation," *in Proce Spring Symposium on Se Problem Solving Under Incomplete Informatio*n, Pal

[24] Reuters-21578, available at: http://www.daviddlewis.com/resources/testcollections/reuters21578/, 2005.

[25] Vapnik V., *The Nature of Statistical Learning Theory*, Springer, New York, USA, 1995.

[26] Yang Y. and Pedersen J. P., "A Comparative Study on Feature Selection in Text Categorization," *in Proceedings of the 4th International Conference on Machine Learning*, Nashville, USA, pp. 412-420, 1997.

Hisham Al-Mubaid obtained his PhD degree in computer science from the University of Texas at Dallas, USA, in 2000. He worked one year as an assistant professor at State University of New York (SUNY), USA. He joined the University of Houston-Clear Lake, USA, in 2001 as an assistant professor of computer science. His research interests and publications have been primarily centered around natural language processing, and include text categorization, machine learning, text mining, semantics and ontology. He also has interests and publications in bioinformatics and teaching-learning research. He serves in the technical and program committees of several journals and conferences.