

Natural Language Interface Models for Fast Responsiveness Applications

Hisham Al-Mubaid

Abstract—A fast responsiveness system incurs minimum latency and produces high throughput and quick response. Interacting with these systems using friendly natural language interfaces requires natural language (NL) capability and an NL processing component. Examples of fast responsive systems include mission-critical systems like aerospace applications and real-time text messaging applications. The NL component constitutes an important part of the interface. We propose NL interface models for fast responsiveness systems with a language disambiguation component. The language disambiguation component is based on using supervised Machine Learning (ML) techniques. For that, we designed and implemented a number of language disambiguation techniques for word prediction to be used with such systems that require fast responsiveness. In the experimental evaluation, proposed techniques demonstrated impressive performance in prediction accuracy.

A FAST RESPONSIVENESS SYSTEM is a system that incurs minimum latency and produces high throughput and quick responses. Interacting with such systems requires smart interfaces with capabilities to maintain and support fast responsiveness. The smart interfaces of these systems require natural language (NL) capabilities and a natural language processing (NLP) component. Examples of fast responsiveness systems include Virtual Reality (VR) training programs, real-time text messaging applications, mission-critical systems like aerospace applications, and applications in which immediate responses are needed.⁵ In general, fast responsiveness is particularly important for (1) mission critical, (2) real time, and (3) aerospace applications.



Dr. Hisham Al-Mubaid

The NLP component constitutes an important part of the interface, but it is the hard part since working with NL will lead to facing the difficult problem of NL ambiguity. In addition, having an *efficient* NL component is very appealing for fast responsiveness applications. We present efficient models of interfaces with NL capabilities for fast responsiveness systems. An important part of the interface is the language disambiguation component. The language disambiguation component is based on integrating adaptive and supervised Machine Learning (ML) techniques. ML techniques have demonstrated outstanding success in many similar problems.³ The basic structure of the pro-

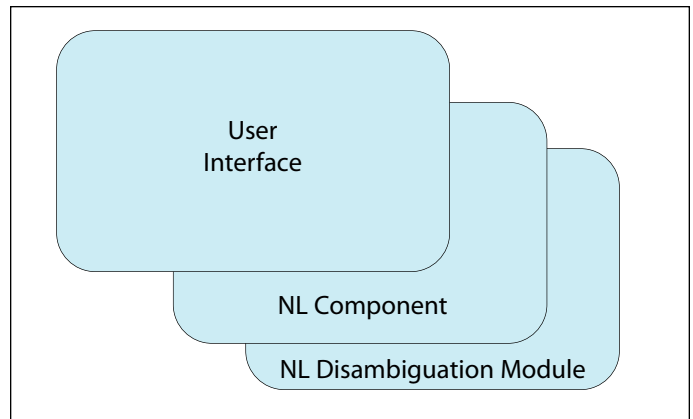


Figure 1. Basic Structure of the Proposed Model

posed design is shown in Fig. 1. The language disambiguation module will be able to perform a number of tasks, including:

- Predicting words and completing user input: the system completes a word being typed by the user; thus, many keystrokes can be saved and text entry delay reduced.
- Correcting real word errors.
- Suggesting words during text entry.

To perform these tasks, we designed and implemented word disambiguation and prediction techniques to solve the NL ambiguity problem. The proposed techniques will allow for word prediction and completion in the NL interfaces of the fast responsiveness applications.

Background and Related Work

One of the major problems in developing robust Natural

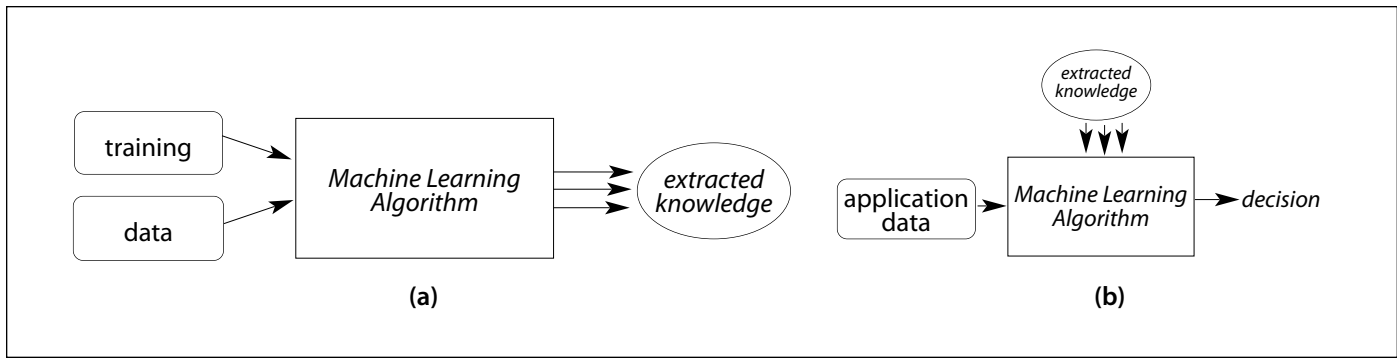


Figure 2. Machine Learning: (a) the Training/Learning Phase (b) the Application Phase

Language Processing (NLP) applications is the *ambiguity* of words and sentences. With the appeal of natural language interfaces, some researchers have argued that a language like English has too many ambiguities to be useful for communicating with computers. Typically, words and phrases have many meanings that can be inferred from the context, and these meanings shift with the context. For example, in word disambiguation, we would like to decide, in a given context, and from a given set of words (*confusion set*) which word is the most likely one in the given context.

The problem of word ambiguity recurs in many applications, and these applications will benefit greatly from an efficient and accurate word disambiguation system.

A number of methods and systems have been developed for the word prediction problem 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 a word prediction task. Fazly presents a comprehensive review of prior related work in word prediction.⁴ An interesting method among related work is one presented by Even-Zohar and Roth.⁷ Their 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. The method was tested in several experiments using for training and testing texts taken from a *Wall Street Journal (WSJ)* corpus.

The Methods

In NLP literature, most of the efficient and robust systems for processing natural language are designed and developed based on learning and training approaches.⁴ Such learning approaches typically require training data and produce *trained models* that can subsequently be employed in the underlying task; see Fig. 2.

This research investigates and explores the process of integrating supervised learning techniques for language disambiguation

to allow for, and assist in, developing effective and robust interfaces for fast responsiveness applications. Moreover, the developed learning models will be tested with smart interfaces with fast responsiveness. For example, one of the features of fast responsiveness is *word prediction and completion* feature.

This research 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 (X²)*. These feature extraction and selection techniques, MI and X², have been used successfully in information retrieval (IR) and text categorization (TC).¹⁰⁻¹² Thus, the WP problem here is cast 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 [$w_n \dots w_3 w_2 w_1$ -?-], we wish to predict and determine the word that follows the sequence [$\dots w_3 w_2 w_1$ (i.e., the word in place of the “-?-”).]. Example of confusion sets used in this research include: {*quite-quiet, peace-piece, passed-past, being-begin, than-then, raise-rise, site-sight*} (Table 1). Now we can summarize the problem as follows. Let $c = \{w_1, w_2, \dots, w_n\}$ be the context of the prediction task, where n as an integer number represents the size of context window (*in this research we tested for n values 3, 5, or 10*). The words w_1, w_2, \dots, w_n are the words that appear immediately before the word to be predicted. Also let $f = \{w_x, w_y\}$ be the confusion set for this case. Our method relies on machine learning to train word classifiers to classify (predict) whether w_x or w_y is the predicted *correct* word in that context. Each word in the confusion set is represented as a projection on the feature vector that is composed from the training data. One of the contributions of this work is in the way in which we extract and compute features from the training data.

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 [$w_n \dots w_3 w_2 w_1 w_x$] or [$w_n \dots w_3 w_2 w_1 w_y$]. 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 exam-

Table 1. Three Confusion Sets Used in the Experiments

Confusion set 1	accept-except, affect-effect, begin-being, country-county,...
Confusion set 2	site-sight, than-then, further-farther, raise-rise,...
Confusion set 3	advice-advise, weak-week, sea-see, lose-loose,...

ple into a feature vector as follows. We select as features only certain words with high “discriminating” capabilities between the two confused words (e.g., w_x and w_y). These features are used to represent each example in training and prediction. 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 for each context word $w_i \in W$, as follows:

- a = number of occurrences of w_i in C_1
- b = number of occurrences of w_i in C_2
- c = number of examples of C_1 that do not contain w_i
- d = number of examples of C_2 that do not contain w_i (0)

Then, the *mutual information* (MI) is defined as:

$$MI = \frac{N \cdot a}{(a + b) \cdot (a + c)} \quad (1)$$

where N is the total number of examples in C_1 and C_2 . Chi-Square (X^2) is computed as:

$$X^2 = \frac{N \cdot (ad - cb)_2}{(a + c) \cdot (b + d) \cdot (a + b) \cdot (c + d)} \quad (2)$$

To give more weight for the difference $(a - b)$ and for the value a , we adapted from *MI* the following two techniques:

$$MI_1 = MI \cdot (a - b) \quad (3)$$

$$MI_2 = MI \cdot a \cdot (a - b) \quad (4)$$

Experiments and Results

Datasets. We used four different text datasets to evaluate our method. Details of the datasets are in Table 2. The datasets are as follows:

- The Reuters is taken from the *Reuters-21578* benchmark dataset.¹³
- The ACL dataset was obtained from the LDC-Linguistic data consortium (<<http://www.ldc.upenn.edu/>>) and includes news stories 1987–1991 from the *WSJ*.
- The BioMed text is a corpus of biomedical articles taken from *Medline*.¹⁴
- The 10-K dataset contains financial text of 10-K filings of U.S. corporations, taken from the U.S. Securities and Exchanges Commission (SEC at <<http://www.sec.gov/>>).

Table 2. Details of the Four Datasets Used in Experiments

Dataset (source)	Training text size words	Testing text size words
Reuters (Reuters-21578)	977,418	167,835
ACL (LDC)	761,730	451,407
Biomed text (<i>Medline</i>)	774,206	466,254
10-K (SEC)	527,390	152,069

Confusion sets. We used three confusion sets in the experiments, shown in Table 2.

Evaluation and Discussion

We used *MI*, *MI_2*, and X^2 for feature selection and *SVM* for learning and prediction; we also used the *naïve Bayes* algorithm¹⁵ as the baseline. For context size, we used preceding 3, 5, or 10 words. Furthermore, for size of the feature vectors, we tried 10, 20, and 30 features and found that the best performance resulted when using 20 features. We initially tested our method using three datasets *Reuters*, *ACL*, and *BioMed* (Table 2), and the three confusion sets (Table 1). The results are presented in Table 3 when using *MI_2* for feature selection and in Table 4 when the X^2 feature selection technique was used. With a total of 19,438, word prediction instances were tested in each experiment (Tables 3 and 4). We noticed that *MI_2* produces slightly better accuracy than X^2 . Moreover, to compare our method against the baseline method (*N. Bayes*) we ran all the testing prediction instances on the *Bayesian* method; the results are in Table 5. The Bayesian method produced slightly better accuracy than *MI_2* only in the *Reuters* dataset, but the other two datasets, *MI_2* and X^2 , outperform *Bayesian* significantly (Table 5). Furthermore, the micro-average accuracy on the three datasets demonstrates that *MI_2* and X^2 outperform the baseline method, as shown in Table 5.

Finally, since the *10-K* dataset is very specialized set and is not as commonly used in NLP as the other datasets, we tested our method on it in a separate experiment using *MI_2* and X^2 ; results in Table 6. 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 6). Overall, this method of learning-classification-based word prediction is capable of achieving accuracy in the range of 87%-88% correct

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

Dataset	Confusion set 1		Confusion set 2		Confusion set 3		Average Accuracy
	num. of tested instances	accuracy	num. of tested instances	accuracy	num. 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
<i>Total</i>	5998		8943		4497		

Table 4. Accuracy results using the three datasets and three confusion sets using X^2 for feature selection, preceding three words for contexts, and top 20 features.

Dataset	Confusion set 1		Confusion set 2		Confusion set 3		Average Accuracy
	num. of tested instances	accuracy	num. of tested instances	accuracy	num. 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
<i>Total</i>	5998		8943		4497		

Table 5. Average accuracy on each method with each dataset; accuracy here is the average of testing on all confusion sets.

Dataset	Num. of tested instances	Accuracy		
		<i>N.Bayes</i>	<i>MI_2</i>	X^2
Reuters	3037	90.67	89.79	87.23
ACL	8176	80.12	85.53	85.12
BioMed	8225	81.28	88.76	87.37
<i>Total</i>	19,438			
Micro. Avg		82.26	87.56	86.40

Table 6. Accuracy Results for the 10-K Dataset

Dataset	Num. of tested instances	Accuracy	
		<i>MI_2</i>	X^2
10-K	2,610	91.42	87.09

predictions using only the three preceding words as context, which emphasizes the robustness of the feature selection techniques and the learning method.

Furthermore, 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 4), and the average accuracy on *Reuters* is approaching ~90%; *BioMed*, approaching ~89% (Table 3). In addition, the method achieved accuracy of 95.2% on the *Reuters* using confusion set 3 (Table 3) and 93.1% on the *BioMed* dataset using confusion set 3 (Table 4).

Conclusion

This paper presents new word disambiguation and prediction techniques to solve the NL ambiguity problem. Proposed techniques were evaluated extensively and demonstrated impressive performance in prediction accuracy. The proposed techniques will allow for word prediction and completion in the NL interfaces of the fast responsiveness applications. Thus, the resulting interface will have attractive features such as predicting and completing words during text entry, correcting real-word misspellings typed by the user, and determining the correct word for

a given sound in speech interface (speech recognition). This will facilitate the interaction and interfacing with the system by effectively speeding up interaction with the computer. These features are highly appealing for mission-critical applications and domains where fast responsiveness is needed, as in aerospace applications. Moreover, word prediction is a very important task and has many significant applications. Besides regular users, a robust word prediction system can benefit able-bodied users by allowing higher text entry rates and minimizing number of typographical errors and misspellings. This aspect has been observed by the developers of the open-source word processor OpenOffice,¹⁶ which provides, along with standard word processing features, word completion.

References

- ¹H. Al-Mubaid and M. Siddiqui, "Automatic Text Categorization with Learning Logic," *Proc.*, 6th Intl. Conf. for Computer Applications in Industry and Engineering, Las Vegas, NV, Nov. 2003.
- ²H. Al-Mubaid, "Context-Based Word Prediction and Classification," *Proc.*, ISCA Intl. Conf. on Computers and Their Applications, 2003.
- ³H. Al-Mubaid and K. Truemper, "Learning To Find Context-Based Spelling Errors," in *Data Mining and Knowledge Discovery Approaches Based on Rule Induction Techniques*, Eds. E. Triantaphyllou and G. Felici. N.Y.: Kluwer Acad. Pub., 2005.
- ⁴A. Fazly, "The Use of Syntax in Word Completion Utilities," Master's thesis, U. of Toronto, Canada, 2002.
- ⁵C. I. Guinn and R. Jorge Montoya, "Natural Language Processing in Virtual Reality Training Environments," *Modern Simulation and Training* (June 1998): 44-55.
- ⁶Y. Even-Zohar and D. Roth, "A Classification Approach to Word Prediction," NAACL (2000).
- ⁷G. Forman, "An Extensive Empirical Study of Feature Selection Metrics for Text Classification," *JMLR* 3 (2003): 1289-305.
- ⁸L. Galavotti, F. Sebastiani, and M. Simi, "Experiments on the Use of Feature Selection and Negative Evidence in Automated Text Categorization," *Proc.*, 4th European Conf. on Research and Advanced Technology for Digital Libraries, 2000.
- ⁹Y. Yang, and J. P. Pedersen, "A Comparative Study on Feature Selection in Text Categorization," in The 4th Intl. Conf. on Machine Learning, Ed. Jr. D. H. Fisher, 1997. 412-20.
- ¹⁰Reuters-21578 Text Categorization Test Collection, May, 14

(Continued on page 94.)



Natural Language Interface Models for Fast Responsiveness Applications

(Continued from page 44.)

2004, David Lewis, Jan. 2006 <<http://www.daviddlewis.com/resources/testcollections/reuters21578/>>.

¹⁴Medline, accessed using Entrez PubMed Interface <<http://www.ncbi.nlm.nih.gov/entrez/query.fcgi>>.

¹⁵C. D. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*. Cambridge: MIT Press, 1999.

¹⁶OpenOffice: A Multiplatform and Multilingual Office Suite and Open-Source Project, Sun Microsystems, 2000 <<http://www.openoffice.org/>>.

Publications

Al-Mubaid, H. "A Learning-Classification Based Approach for Word Prediction," *IAJIT Journal*, 2006. (Under review.)

—. "Context-Based Technique for Biomedical Term Classification," IEEE GrC-06, 2006. (Submitted, under review.)

Al-Mubaid, H. and K. Truemper. "Learning To Find Context-Based Spelling Errors," in *Data Mining and Knowledge Discovery Approaches Based on Rule Induction Techniques*. Eds. E. Triantaphyllou and G. Felici. N.Y.: Kluwer Acad. Pub., 2005.

Al-Mubaid, H. and R. Singh. "A New Text Mining Approach for Finding Protein-to-Disease Associations," *Am. J. Biochem. Biotechnol.* 2.2 (2005).

Presentations

Al-Mubaid, H. "Context-Based Similar Words Detection and Its Application in Specialized Search Engines," Intl Conf. on Intelligent User Interfaces, San Diego, CA, 2005.

—. "Machine Learning Approach for Context Sensitive Error Detection," Intl. Conf. on Intelligent Computing and Information Systems, Cairo, Egypt, 2005.

Funding and Proposals

"Integrating Supervised and Adaptive Learning to Improve Computer Accessibility," NSF, CIS, Information and Intelligent Systems IIS, request for two years, \$164,000. (Submitted).

COMET HALE-BOPP—Comet Hale-Bopp in the constellation Andromeda was photographed by George Shelton, photographer for the Bionetics Corp., at 8:14 p.m. on March 31, 1997, from Merritt Island, Florida, close to the Kennedy Space Center. During that 24-hour period, Comet Hale-Bopp made its closest approach to the Sun.