10-2005

NIL Is not nothing: Recognition of Chinese network informal language expressions

Yunqing XIA

Kam-Fai WONG

Wei GAO
Singapore Management University, weigao@smu.edu.sg

NIL Is Not Nothing: Recognition of Chinese Network Informal Language Expressions

Yunqing Xia, Kam-Fai Wong, Wei Gao
Department of Systems Engineering and Engineering Management
The Chinese University of Hong Kong, Shatin, Hong Kong
{yqxia, kfwong, wgao}@se.cuhk.edu.hk

Abstract

Informal language is actively used in network-mediated communication, e.g. chat room, BBS, email and text message. We refer the anomalous terms used in such context as network informal language (NIL) expressions. For example, "偶(ou3)" is used to replace "我(wo3)" in Chinese ICQ. Without unconventional resource, knowledge and techniques, the existing natural language processing approaches exhibit less effectiveness in dealing with NIL text. We propose to study NIL expressions with a NIL corpus and investigate techniques in processing NIL expressions. Two methods for Chinese NIL expression recognition are designed in NILER system. The experimental results show that pattern matching method produces higher precision and support vector machines method higher F-1 measure. These results are encouraging and justify our future research effort in NIL processing.

1 Introduction

The rapid global proliferation of Internet applications has been showing no deceleration since the new millennium. For example, in commerce more and more physical customer services/call centers are replaced by Internet solutions, e.g. via MSN, ICQ, etc. Network informal language (NIL) is actively used in these applications. Following this trend, we forecast that NIL would become a key language for human communication via network.

Today NIL expressions are ubiquitous. They appear, for example, in chat rooms, BBS, email, text message, etc. There is growing importance in understanding NIL expressions from both technology and humanity research points of view. For instance, comprehension of customer-operator dialogues in the aforesaid commercial application would facilitate effective Customer Relationship Management (CRM).

Recently, sociologists showed many interests in studying impact of network-mediated communication on language evolution from psychological and cognitive perspectives (Danet, 2002; McElhearn, 2000; Nishimura, 2003). Researchers claim that languages have never been changing as fast as today since inception of the Internet; and the language for Internet communication, i.e. NIL, gets more concise and effective than formal language.

Processing NIL text requires unconventional linguistic knowledge and techniques. Unfortunately, developed to handle formal language text, the existing natural language processing (NLP) approaches exhibit less effectiveness in dealing with NIL text. For example, we use ICTCLAS (Zhang et al., 2003) tool to process sentence "他细八细要开会啊？(Is he going to attend a meeting?)". The word segmentation result is "他|细|八|细|要|开会|啊|？ ". In this sentence , "细八细(xi4 ba1 xi4)" is a NIL expression which means 'is he ….?' in this case. It can be concluded that without identifying the expression, further Chinese text processing techniques are not able to produce reasonable result.

This problem leads to our recent research in "*NIL is Not Nothing*" project, which aims to produce techniques for NIL processing, thus avails understanding of change patterns and behaviors in language (particularly in Internet language) evolution. The latter could make us more adaptive to the dynamic language environment in the cyber world.

Recently some linguistic works have been carried out on NIL for English. A shared dictionary

has been compiled and made available online. It contains 308 English NIL expressions including English abbreviations, acronyms and emoticons. Similar efforts for Chinese are rare. This is because Chinese language has not been widely used on the Internet until ten years ago. Moreover, Chinese NIL expression involves processing of Chinese Pinyin and dialects, which results in higher complexity in Chinese NIL processing.

In "*NIL is Not Nothing*" project, we develop a comprehensive Chinese NIL dictionary. This is a difficult task because resource of NIL text is rather restricted. We download a collection of BBS text from an Internet BBS system and construct a NIL corpus by annotating NIL expressions in this collection by hand. An empirical study is conducted on the NIL expressions with the NIL corpus and a knowledge mining tool is designed to construct the NIL dictionary and generate statistical NIL features automatically. With these knowledge and resources, the NIL processing system, i.e. NILER, is developed to extract NIL expressions from NIL text by employing state-of-the-art information extraction techniques.

The remaining sections of this paper are organized as follow. In Section 2, we observe formation of NIL expressions. In Section 3 we present the related works. In Section 4, we describe NIL corpus and the knowledge engineering component in NIL dictionary construction and NIL features generation. In Section 5 we present the methods for NIL expression recognition. We outline the experiments, discussions and error analysis in Section 6, and finally Section 7 concludes the paper.

2 The Ways NIL Expressions Are Typically Formed

NIL expressions were first introduced for expediting writing or computer input, especially for online chat where the input speed is crucial to prompt and effective communication. For example, it is rather annoying to input full Chinese sentences in text-based chatting environment, e.g. over the mobile phone. Thus abbreviations and acronyms are then created by forming words in capital with the first letters of a series of either English words or Chinese Pinyin.

Chinese Pinyin is a popular approach to Chinese character input. Some Pinyin input methods incorporate lexical intelligence to support word or phrase input. This improves input rate greatly. However, Pinyin input is not error free. Firstly, options are usually prompted to user and selection errors result in homophone, e.g. "斑竹 (ban1 zu2)" and "版主 (ban1 zhu3)". Secondly, input with incorrect Pinyin or dialect produces wrong Chinese words with similar pronunciation, e.g. "稀饭 (xi1 fan4)" and "喜欢 (xi3 hua-n1)". Nonetheless, prompt communication spares little time to user to correct such a mistake. The same mistake in text is constantly repeated, and the wrong word thus becomes accepted by the chat community. This, in fact, is one common way that a new Chinese NIL expression is created.

We collect a large number of "sentences" (strictly speaking, not all of them are sentences) from a Chinese BBS system and identify NIL expressions by hand. An empirical study on NIL expressions in this collection shows that NIL expressions can be classified into four classes as follow based on their origins.

1) Abbreviation (A). Many Chinese NIL expressions are derived from abbreviation of Chinese Pinyin. For example, "PF" equals to "佩服 (pei4 fu2)" which means "admire".
2) Foreign expression (F). Popular Informal expressions from foreign languages such as English are adopted, e.g. "ASAP" is used for "as soon as possible".
3) Homophone (H). A NIL expression is sometimes generated by borrowing a word with similar sound (i.e. similar Pinyin). For example "稀饭" equals "喜欢" which means "like". "稀饭" and "喜欢" hold homophony in a Chinese dialect.
4) Transliteration (T) is a transcription from one alphabet to another and a letter-for-letter or sound-for-letter spelling is applied to represent a word in another language. For example, "拜拜 (bai4 bai4)" is transliteration of "bye-bye".

A thorough observation, in turn, reveals that, based on the ways NIL expressions are formed and/or their part of speech (POS) attributes, we observe a NIL expression usually takes one of the forms presented in Table 1 and Table 2.

The above empirical study is essential to NIL lexicography and feature definition.

Table 1: NIL expression forms based on word formation.

Word Formation	# of NIL Expressions	Examples
Chinese Word or Phrase	33	"稀饭" represents "喜欢" and means "like".
Sequence of English Capitals	341	"PF" represents "佩服" and means "admire".
Number	8	"94 (jiu3 si4)" represents "就是 (jiu4 shi4)" and means "exactly be".
Mixture of the Above Forms	30	"8 错 (ba1 cuo4)" represents "不错 (bu3 cuo4)" and means "not bad".
Emoticons	239	":-(" represents a sad emotion.

Table 2: NIL expression forms based on POS attribute.

POS Attribute	# of NIL Expressions	Examples
Number	1	"W" represents "万 (wan4)" and means "ten thousand".
Pronoun	9	"偶" represents "我" and means "I".
Noun	29	"LG" represents "老公 (lao3 gong1)" and means "husband".
Adjective	250	"FB" represents "腐败 (fu3 bai4)" and means "corrupt".
Verb	34	"葱白 (cong1 bai2)" represents "崇拜 (chong3 bai4)" and means "adore".
Adverb	10	"粉 (fen3)" represents "很 (hen3)" and means "very".
Exclamation	9	"捏 (nie0)" represents "呢 (ne0)" and equals a descriptive exclamation.
Phrase	309	"AFK" represents "Away From Keyboard".

3 Related Works

NIL expression recognition, in particular, can be considered as a subtask of information extraction (IE). Named entity recognition (NER) happens to hold similar objective with NIL expression recognition, i.e. to extract meaningful text segments from unstructured text according to certain predefined criteria.

NER is a key technology for NLP applications such as IE and question & answering. It typically aims to recognize names for person, organization, location, and expressions of number, time and currency. The objective is achieved by employing either handcrafted knowledge or supervised learning techniques. The latter is currently dominating in NER amongst which the most popular methods are decision tree (Sekine et al., 1998; Pailouras et al., 2000), Hidden Markov Model (Zhang et al., 2003; Zhao, 2004), maximum entropy (Chieu and Ng, 2002; Bender et al., 2003), and support vector machines (Isozaki and Kazawa, 2002; Takeuchi and Collier, 2002; Mayfield, 2003).

From the linguistic perspective, NIL expressions are rather different from named entities in nature. Firstly, named entity is typically noun or noun phrase (NP), but NIL expression can be any kind, e.g. number "94" in NIL represents "就是" which is a verb meaning "exactly be". Secondly, named entities often have well-defined meanings in text and are tractable from a standard dictionary; but NIL expressions are either unknown to the dictionary or ambiguous. For example, "稀饭" appears in conventional dictionary with the meaning of Chinese porridge, but in NIL text it represents "喜欢" which surprisingly represents "like". The issue that concerns us is that these expressions like "稀饭" may also appear in NIL text with their formal meaning. This leads to ambiguity and makes it more difficult in NIL processing.

Another notable work is the project of "*Normalization of Non-standard Words*" (Sproat et al., 2001) which aims to detect and normalize the "*Non-Standard Words* (NSW)" such as digit sequence; capital word or letter sequence; mixed case word; abbreviation; Roman numeral; URL and e-mail address. In our work, we consider most types of the NSW in English except URL and email address. Moreover, we consider Chinese NIL expressions that contain same characters as the normal words. For example, "稀饭" and "葱白" both appear in common dictionaries, but they carry anomalous meanings in NIL text. Ambiguity arises and basically brings NIL expressions recognition beyond the scope of NSW detection.

According to the above observations, we propose to employ the existing IE techniques to handle NIL expressions. Our goal is to develop a NIL expression recognition system to facilitate network-mediated communication. For this purpose, we first construct the required NIL knowledge resources, namely, a NIL dictionary and n-gram statistical features.

4 Knowledge Engineering

Recognition of NIL expressions relies on unconventional linguistic knowledge such as NIL dictionary and NIL features. We construct a NIL corpus and develop a knowledge engineering component to obtain these knowledge by running a knowledge mining tool on the NIL corpus. The knowledge mining tool is a text processing program that extracts NIL expressions and their attributes and contextual information, i.e. n-grams, from the NIL corpus. Workflow for this component is presented in Figure 1.

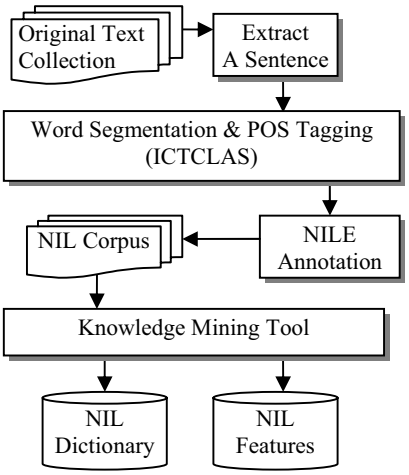

Figure 1: Workflow for NIL knowledge engineering component. NILE refers to NIL expression, which is identified and annotated by human annotator.

4.1 NIL Corpus

The NIL corpus is a collection of network informal sentences which provides training data for NIL dictionary and statistical NIL features. The NIL corpus is constructed by annotating a collection of NIL text manually.

Obtaining real chat text is difficult because of the privacy restriction. Fortunately, we find BBS text within "大嘴区(da4 zui3 qu1)" zone in YESKY system (http://bbs.yesky.com/bbs/) reflects remarkable colloquial characteristics and contains a vast amount of NIL expressions. We download BBS text posted from December 2004 and February 2005 in this zone. Sentences with NIL expressions are selected by human annotators, and NIL expressions are manually identified and annotated with their attributes. We finally collected 22,432 sentences including 451,193 words and 22,648 NIL expressions.

The NIL expressions are marked up with SGML. The typical example, i.e. "他细八细要开会啊？" in Section 1, is annotated as follows.

他<*NILEX string*="细八细" *class*="H" *normal*="是不是" *pinyin*="xi4 ba1 xi4" *segments*="细|八|细" *pos*="VERB" *posseg*="ADJ|NUM|ADJ">细八细</*NILEX*>要开会啊？

where *NILEX* is the SGML tag to label a NIL expression, which entails NIL linguistic attributes including *class, normal, pinyin, segments, pos,* and *posseg* (see Section 4.2). *H* is a value of *class* (see Section 2). Value *VERB* demotes verb, ADJ adjective, NUM number and *AUX* auxiliary.

4.2 NIL Dictionary

The NIL dictionary is a structured databank that contains NIL expression entries. Each entity in turn entails nine attributes described as follow.

1. *ID*: an unique identification number for the NIL expression, e.g. 915800;
2. *string*: string of the NIL expression, e.g. "细八细";
3. *class*: class of the NIL expression (see Section 2), e.g. "H" for homophony;
4. *pinyin*: Chinese Pinyin for the NIL expression, e.g. "xi4 ba1 xi4";
5. *normal*: corresponding normal text for the NIL expression, e.g. "是不是";
6. *segments*: word segments of the NIL expression, e.g. "细|八|细";
7. *pos*: POS tag associated with the expression, e.g. "VERB" denoting a verb;
8. *posseg*: a POS tag list for the word segments, e.g. "VERB|AUX|VERB";
9. *frequency*: number of occurrences of the NIL expression.

We run the knowledge mining tool to extract all annotated NIL expressions together with their attributes from the NIL corpus. The NIL expressions are then each assigned an ID number and inserted into an indexed data file, i.e. the NIL dictionary. Current NIL dictionary contains 651 NIL entries.

4.3 NIL Feature Set

The NIL features are required by support vector machines method in NIL expression recognition. We define two types of statistical features for NIL expressions, i.e. Chinese word n-grams and POS tag n-grams. Bigger *n* leads to more contextual

information, but results in higher computational complexity. To compromise, we generate n-grams with n = 1, 2, 3, 4. For example, "你们/细八细" is a bi-gram for "细八细" in terms of word segmentation, and its POS tag bi-gram is "PRONOUN/ VERB".

We run the knowledge mining tool on the NIL corpus to produce all n-grams for Chinese words and their POS tags in which NIL expression appears. 8379 features were generated including 7416 word-based n-grams and 963 POS tag-based n-grams. These statistical NIL features are linked to the corresponding NIL dictionary entries by their global NIL expression IDs.

Besides, we consider some morphological features including being/containing a number, some English capitals or Chinese characters. These features can be extracted by parsing string of the NIL expressions.

5 NILER System

5.1 Architecture

We develop NILER system to recognize NIL expressions in NIL text and convert them to normal language text. The latter functionality is discussed in other literatures. Architecture of NILER system is presented in Figure 2.

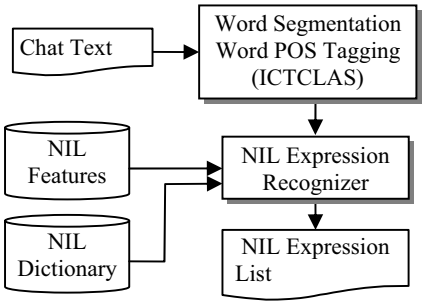

Figure 2: Architecture of NILER system.

The input chat text is first segmented and POS tagged with ICTCLAS tool. Because ICTCLAS is not able to identify NIL expressions, some expressions are broken into several segments. NIL expression recognizer processes the segments and POS tags and identifies the NIL expressions.

5.2 NIL Expression Recognizer

We implement two methods in NIL expression recognition, i.e. pattern matching and support vector machines.

5.2.1 Method I: Pattern Matching

Pattern matching (PM) is a traditional method in information extraction systems. It uses a handcrafted rule set and dictionary for this purpose. Because it's simple, fast and independent of corpus, this method is widely used in IE tasks.

By applying NIL dictionary, candidates of NIL expressions are first extracted from the input text with longest matching. As ambiguity occurs constantly, 24 patterns are produced and employed to disambiguate. We first extract those word and POS tag n-grams from the NIL corpus and create patterns by generalizing them manually. An illustrative pattern is presented as follows.

$$[< v\_any >]\ 8 < not(v\_unit) > [< v\_any >] \Rightarrow 不$$

where v_any and v_unit are variables denoting any word and any unit word respectively; $not(\bullet)$ is the negation operator. The illustrative pattern determines "8" to be a NIL expression if it is succeeded by a unit word. With this pattern, "8" within sentence "他工作了 8 个小时。 (He has been working for eight hours.)" is not recognized as a NIL expression.

5.2.2 Method II: Support Vector Machines

Support vector machines (SVM) method produces high performance in many classification tasks (Joachims, 1998; Kudo and Matsumoto, 2001). As SVM can handle large numbers of features efficiently, we employ SVM classification method to NIL expression recognition.

Suppose we have a set of training data for a two-class classification problem $\{(x_1,y_1), (x_2, y_2),\ldots,(x_N, y_N)\}$, where $x_i \in R^D (i = 1,2,...N)$ is a feature vector of the i-th order sample in the training set and $y_i \in \{+1,-1\}$ is the label for the sample. The goal of SVM is to find a decision function that accurately predicts y for unseen x. A non-linear SVM classifier gives a decision function $f(x) = sign(g(x))$ for an input vector x, where

$$g(x) = \sum_{i=1}^{l} \varpi_i K(x, z_i) + b$$

The $z_i s$ are so-called support vectors, and represents the training samples. ϖ_i and b are parameters for SVM motel. l is number of training samples. $K(x,z)$ is a kernel function that implic-

itly maps vector x into a higher dimensional space. A typical kernel is defined as dot products, i.e. $K(x,z)=k(x\bullet z)$.

Based on the training process, the SVM algorithm constructs the support vectors and parameters. When text is input for classification, it is first converted into feature vector x. The SVM method then classifies the vector x by determining sign of $g(x)$, in which $f(x)=1$ means that word x is positive and otherwise if $f(x)=-1$. The SVM algorithm was later extended in $SVM^{multiclass}$ to predict multivariate outputs (Joachims, 1998).

In NIL expression recognition, we consider NIL corpus as training set and the annotated NIL expressions as samples. NIL expression recognition is achieved with the five-class SVM classification task, in which four classes are those defined in Section 2 and reflected by *class* attribute within NIL annotation scheme. The fifth class is *NOCLASS*, which means the input text is not any NIL expression class.

6 Experiments

6.1 Experiment Description

We conduct experiments to evaluate the two methods in performing the task of NIL expression recognition. In training phase we use NIL corpus to construct NIL dictionary and pattern set for PM method, and generate statistical NIL features, support vectors and parameters for SVM methods. To observe how performance is influenced by the volume of training data, we create five NIL corpora, i.e. C#1~C#5, with five numbers of NIL sentences, i.e. 10,000, 13,000, 16,000, 19,000 and 22,432, by randomly selecting sentence from NIL corpus described in Section 4.1.

To generate test set, we download 5,690 sentences from YESKY system which cover BBS text in March 2005. We identify and annotate NIL expressions within these sentences manually and consider the annotation results as gold standard.

We first train the system with the five corpora to produce five versions of NIL dictionary, pattern set, statistical NIL feature set and SVM model. We then run the two methods with each version of the above knowledge over the test set to produce recognition results automatically. We compare these results against the gold stand and present experimental results with criteria including precision, recall and F1-measure.

6.2 Experimental Results

We present experimental results of the two methods on the five corpora in Table 3.

Table 3: Experimental results for the two methods on the five corpora. PRE denotes precision, REC denotes recall, and F1 denotes F1-Measure.

Corpus	PM			SVM		
	PRE	REC	F1	PRE	REC	F1
C#1	0.742	0.547	0.630	0.683	0.703	0.693
C#2	0.815	0.634	0.713	0.761	0.768	0.764
C#3	0.873	0.709	0.783	0.812	0.824	0.818
C#4	0.904	0.759	0.825	0.847	0.851	0.849
C#5	0.915	0.793	0.850	0.867	0.875	0.871

6.3 Discussion I: The Two Methods

To compare performance of the two methods, we present the experimental results with smoothed curves for precision, recall and F1-Mesure in Figure 3, Figure 4 and Figure 5 respectively.

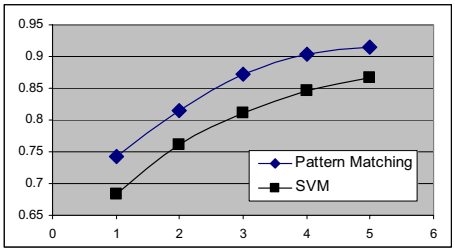

Figure 3: Smoothed precision curves over the five corpora.

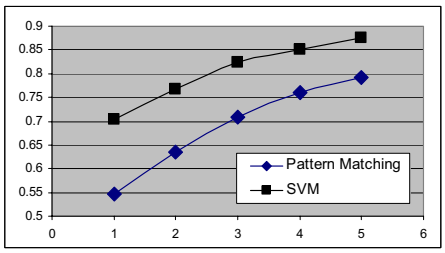

Figure 4: Smoothed recall curves over the five corpora.

Figure 3 reveals that PM method produces higher precision, i.e. 91.5%, and SVM produces higher recall, i.e. 79.3%, and higher F1-Measure, i.e. 87.1%, with corpus C#5. It can be inferred that PM method is self-restrained. In other words, if a NIL expression is identified with this method, it is very likely that the decision is right. However, the weakness is that more NIL expressions are neglected. On the other hand, SVM method outper-

forms PM method regarding overall capability, i.e. F1-Measure, according to Figure 5.

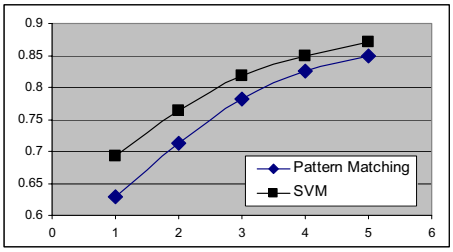

Figure 5: Smoothed F1-Measure curves over the five corpora.

We argue that each method holds strength and weakness. Different methods should be adopted to cater to different application demands. For example, in CRM text processing, we might favor precision. So PM method may be the better choice. On the other hand, to perform the task of chat room security monitoring, recall is more important. Then SVM method becomes the better option. We claim that there exists an optimized approach which combines the two methods and yields higher precision and better robustness at the same time.

6.4 Discussion II: How Volume Influences Performance

To observe how training corpus influences performance in the two methods regarding volume, we present experimental results with smoothed quality curves for the two method in Figure 6 and Figure 7 respectively.

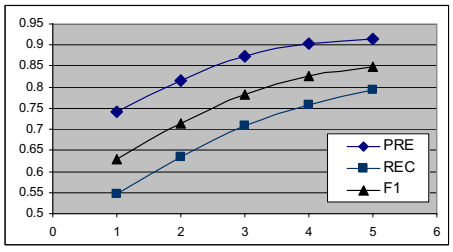

Figure 6: Smoothed quality curves for PM method over the five corpora.

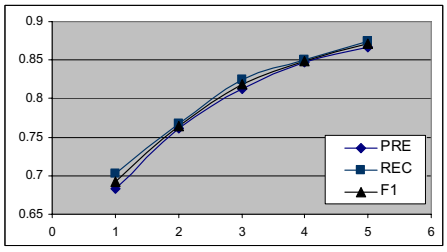

Figure 7: Smoothed quality curves for SVM method over the five corpora.

The smoothed quality curves in Figure 6 and Figure 7 reveal the tendency that bigger volume of training data leads to better processing quality. Meanwhile, the improvement tends to decrease along with increasing of volume. It thus predicts that there exists a corpus with a certain volume that produces the best quality according to the tendency. Although current corpus is not big enough to prove the optimal volume, the tendency revealed by the curves is obvious.

6.5 Error Analysis

We present two examples to analyze errors occur within our experiments.

Err.1 Ambiguous NIL Expression
Example 1:
[Sentence]: 我还是 8 米白
[Meaning]: I still don't understand.
[NIL expression found(Y/N)?]: Y
[Normal language text]: 我还是不明白

Error in Example 1 is caused by failure in identifying "米白 (mi3 bai2)". Because "米 (mi3)" succeeds "8(ba1)" in the word segments, i.e. "我|还是|8|米|白", and it can be used as a unit word, PM method therefore refuses to identify "8(ba)" as a NIL expression according to the pattern described in Section 5.2.1. In fact, "米白" is an unseen NIL expression. SVM method successfully recognizes "米白" to be "米有(mi3 you3)", thus recognizes "8". In our experiments 56 errors in PM method suffer the same failure, while SVM method identifies 48 of them. This demonstrates that PM method is self-restrained and SVM method is relatively scalable in processing NIL text.

Err.2 Unseen NIL expression
Example 2:
[Sentence]: 刚刚从 4U 回来
[Meaning]: Just came back from 4U.
[NIL expression found (Y/N)?] : N

Actually, there is no NIL expression in example 2. But because of a same 1-gram with "4D", i.e. "4", SVM outputs "4U" as a NIL expression. In fact, it is the name for a mobile dealer. There are 78 same errors in SVM method in our experiments, which reveals that SVM method is sometimes over-predicting. In other words, some NIL expressions are recognized with SVM method by mistake, which results in lower precision.

7 Conclusions and Future Works

Network informal language processing is a new NLP research application, which seeks to recognize and normalize NIL expressions automatically in a robust and adaptive manner. This research is crucial to improve capability of NLP techniques in dealing with NIL text. With empirical study on Chinese network informal text and NIL expressions, we propose two NIL expression recognition methods, i.e. pattern matching and support vector machines. The experimental results show that PM method produces higher precision, i.e. 91.5%, and SVM method higher F-1 measure, i.e. 87.1%. These results are encouraging and justify our future research effort in NIL processing.

Research presented in this paper is preliminary but significant. We address future works as follow. Firstly, NIL corpus constructed in our work is fundamental. Not only will difficulty in seeking for text resource be overcome, but a large quantity of manpower will be allocated to this laborious and significant work. Secondly, new NIL expressions will appear constantly with booming of network-mediated communication. A powerful NIL expression recognizer will be designed to improve adaptivity of the recognition methods and handle the unseen NIL expressions effectively. Finally, we state that research in this paper targets in special at NIL expressions in China mainland. Due to cultural/geographical variance, NIL expressions in Hong Kong and Taiwan could be different. Further research will be conducted to adapt our methods to other NIL communities.

References

Bender, O., Och, F. J. and Ney, H. 2003. *Maximum Entropy Models for Named Entity Recognition,* CoNLL-2003, pp. 148-151.

Chieu, H. L. and Ng, H. T. 2002. *Named Entity Recognition: A Maximum Entropy Approach Using Global Information*. COLING-02, pp. 190-196.

Danet, B. 2002. *The Language of Email*, European Union Summer School, University of Rome.

Isozaki, H. and Kazawa, H. 2002. *Efficient Support Vector Classifiers for Named Entity Recognition*, COLING-02, pp. 390-396..

Joachims, T. 1998. *Text categorization with Support Vector Machines: Learning with many relevant features*. ECML'98, pp. 137-142.

Kudo, T. and Matsumoto, Y. 2001. *Chunking with Support Vector Machines*. NAACL 2001, pp.192-199.

Mayfield, J. 2003. Paul McNamee; Christine Piatko, *Named Entity Recognition using Hundreds of Thousands of Features*, CoNLL-2003, pp. 184-187.

McElhearn, K. 2000. *Writing Conversation - An Analysis of Speech Events in E-mail Mailing Lists*, http://www.mcelhearn.com/cmc.html, Revue Française de Linguistique Appliquée, volume V-1.

Nishimura, Y. 2003. *Linguistic Innovations and Interactional Features of Casual Online Communication in Japanese*, JCMC 9 (1).

Pailouras, G., Karkaletsis, V. and Spyropoulos, C. D. 2000. *Learning Decision Trees for Named-Entity Recognition and Classification*. Workshop on Machine Learning for Information Extraction, ECAI(2000).

Sekine, S., Grishman, R. and Shinnou, H. 1998. *A Decision Tree Method for Finding and Classifying Names in Japanese Texts*, WVLC 98.

Snitt, E. N. 2000. *The Use of Language on the Internet*, http://www.eng.umu.se/vw2000/Emma/linguistics1.htm.

Sproat, R., Black, A., Chen, S., Kumar, S., Ostendorf, M. and Richards, M. 2001. *Normalization of Non-standard Words*. Computer Speech and Languages, 15(3):287- 333.

Takeuchi, K. and Collier, N. 2002. *Use of Support Vector Machines in Extended Named Entity Recognition*. CoNLL-2002, pp. 119-125.

Zhang, Z., Yu, H., Xiong, D. and Liu, Q. 2003. *HMM-based Chinese Lexical Analyzer ICTCLAS*. In the 2nd SIGHAN workshop affiliated with ACL'03, pp. 184-187.

Zhao, S. 2004. *Named Entity Recognition in Biomedical Texts Using an HMM model*, COLING-04 workshop on Natural Language Processing in Biomedicine and its Applications.