

A Two-Stage Incremental Annotation Approach to Constructing A Network Informal Language Corpus

Prof. Kam-Fai Wong
The Chinese University of Hong Kong



A few questions about NIL...

- What is **Network Informal Language**?
- Why does NIL **attract our attention**?
- Why is NIL **special**?
- Why is NIL corpus **necessary**?



What is Network Informal Language?

- **Network Informal Language (NIL)** refers to
 - the special human language widely used in the community of digital network chat via platforms such as chat rooms/tools, mobile phone short message ser-vices (SMS), bulletin board systems (BBS), emails, etc.
- Two examples:
 - “Did we meet **b4**?”
 - b4=before
 - “94他对偶无理” (It is exactly him who is rude to me.)
 - 94(*jiu3 shi4*, ninety four)= 就是(*jiu4 shi4*, exactly be)
 - 偶(*ou3*, even – opposite to odd)=我(*wo3*, me)



Why does NIL attract our attention?

- NIL is ubiquitous in numerous Internet solutions such as:
 - Online support and CRM
 - online education
- NIL is also widely used in online BBS and chat rooms where the facilities are abused by solicitors of
 - terrorism, pornography and crime
- Chat-style NIL is popular in mobile text message, for example,
 - Germans send 200 million text messages a year



Why is NIL special?

- NIL holds anomalous characteristics in forming non-alphabetical characters, words, and phrases
 - "b4" replaces "before"
 - "94(jiu3 si4)" replaces "就是(jiu4 shi4, exactly be)"
- The anomaly brings troubles to conventional NLP tools, e.g.
 - word segmentation: "细 8 细(xi4 ba1 xi4)" replaces "是不是(shi4 bu4 shi 4, be or be not)" but is segmented to "细(slim) 8 (eight) 细"
 - POS ambiguity: "稀饭(xi1 fan4)" means porridge (noun) normally but replaces "喜欢(xi3 huan1, like)" (verb) in NIL
- Our conclusion:
 - Special tools are urgently needed to recognize and normalized NIL text so that conventional NLP tools could be applied.



Why is NIL corpus **necessary**?

- NIL is highly dynamic
 - new NIL terms and NIL phrasal patterns are created every day
- (Xia & Wong 2005) proves that
 - knowledge based approaches, e.g. pattern matching, are difficult to deal with unseen NIL terms.
 - corpus based machine learning approaches, e.g. SVM, are proved to be more robust to predict unseen NIL terms



Task-oriented annotation

- Our ultimate objectives in creating the NIL corpus is to build tools to
 - extract NIL terms from chat text
 - convert these terms into normal text
- Thus the tasks are
 - NIL term recognition and
 - NIL term normalization



NILEML annotation scheme

- NILEML: <NILEX attributes> 细 8 细 </NILEX>

```
attributes ::= nid string class normal pos  
[segments] [posseg] [pinyin]  
nid ::= n<integer>  
string ::= CDATA  
class ::= CDATA  
{class ::= 'A'|'F'|'H'|'T'|'O'}  
normal := CDATA ;  
pos := CDATA  
{pos := 'NOUN'|'PRON'|'VERB'  
|'ADJ'|'ADV'|'NUMBER'|'UNIT'  
|'PREP'|'CONJ'|'AUX'|'EXCL'}  
posseg := CDATA  
{posseg := 'pos|+'}  
pinyin := CDATA ; Chinese pinyin
```



Data source

- Obtaining large scale real chat text is difficult due to privacy concerns.
- We have located BBS chat texts within “大嘴区(da4 zui3 qu1, free chat zone)” in YESKY BBS system
 - December 2004 to July 2005
 - 12,112 pieces of NIL text
 - 92,314 words
 - 12,983 NIL terms.



Annotation tools

- Computer aided annotation platform
 - GUI, interactive (C++, Windows)
 - ICTCLAS tool (ICT/CAS): word segmentation and POS tagging
 - Chinese Pinyin transcription tool (CEDICT)
- Automated annotation module
 - Binary SVM classifier: trained on the annotated NIL terms and used to identify NIL terms in new chat text automatically
 - Tag duplicator: duplicate NILX tag from existing NIL term to the identified same one
 - Tag creator: create empty tag for non-existing identified NIL term



Features for binary SVM classifier

- Occurrences of NIL term when its
 - string appears in any word bi-grams or tri-grams,
 - POS tag appears in any POS tag bi-grams or tri-grams;
 - POS tags for segments appear in any POS tag bi-grams or tri-grams;
- Boolean values that indicate whether a NIL term
 - is a number (Chinese or Arabic);
 - contains merely Latin capitals;
 - contains more than two standard Chinese words;
 - contains punctuations;
 - mixes Chinese character and number;
 - mixes Chinese character and Latin characters;



The two-stage incremental annotation approach

- Overview:
 - Some chat texts are annotated in the first stage manually
 - An automated annotation module is
 - trained on the annotated chat texts incrementally and
 - applied to annotate the rest chat texts automatically



Stage I: manual annotation

- The raw NIL text pieces are split into several blocks according to timestamp in which each block contains 1000 NIL text pieces.
- the first block of 1000 pieces of raw NIL text are annotated under the specification of NILEML by human annotators on the annotation platform
- ICTCLAS tool is integrated to produce word segmentation and POS tags
- Chinese Pinyin transcription tool is employed to produce standard Chinese Pinyin for Chinese characters.



Stage II: incremental annotation

- A binary SVM classifier is trained on the annotated chat text and deployed to identify chat terms in every next block of 1000 pieces of new chat text.
- The Automated annotation module produces basic annotation for each identified NIL term
- Human annotators are expected to
 - confirm or revise new annotations
 - recognize the suggested unidentified NIL terms and produce their annotations manually
- The incremental iterative annotation is repeated until all chat texts are annotated.



Annotation consistency

- Guiding annotation principles
 - Negotiate between annotators to produce an agreed annotation for each new NIL term
 - Duplicate previous annotation for the NIL expression in existence.
 - Each revision should be agreed by all annotators
 - Each revision should be reflected to all same NIL terms
- Under the above guidelines, a higher inter-annotator agreement is guaranteed



Evaluation

- Training/test data
 - A simulation of the annotation procedure, i.e.
 - The first block of 1000 pieces of chat text are used as first training set and the second block is used as test set in the first round of experiment.
 - In each next round of experiment, the annotated chat texts are used as training set and the next block of 1000 pieces of chat text are used as test set
 - there are 11 rounds of experiments in total.



Training/test data description for the 11 test sets

| Round. No. | # of training NIL terms | # of test NIL terms | # of annotated test NIL terms | # of unannotated test NIL terms |
|------------|-------------------------|---------------------|-------------------------------|---------------------------------|
| 1 | 996 | 997 | 414 | 583 |
| 2 | 1992 | 998 | 494 | 504 |
| 3 | 2989 | 1000 | 564 | 436 |
| 4 | 3988 | 997 | 583 | 414 |
| 5 | 4984 | 1001 | 648 | 353 |
| 6 | 5984 | 998 | 702 | 296 |
| 7 | 6981 | 995 | 713 | 282 |
| 8 | 7975 | 992 | 764 | 228 |
| 9 | 8966 | 998 | 791 | 207 |
| 10 | 9963 | 996 | 861 | 135 |
| 11 | 11956 | 1112 | 999 | 113 |



Experimental results

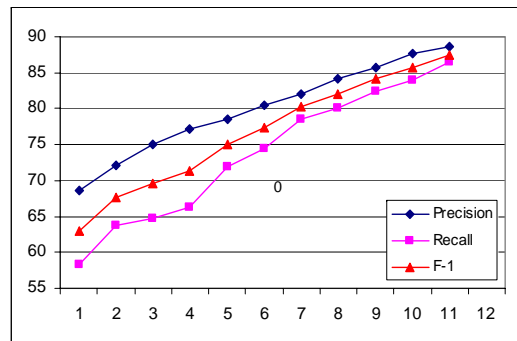
- Overall results for all NIL terms

| Round No. | Precision | Recall | F-1 |
|-----------|-------------|-------------|-------------|
| 1 | 68.6 | 58.3 | 63.0 |
| 2 | 72.1 | 63.8 | 67.7 |
| 3 | 75.1 | 64.7 | 69.5 |
| 4 | 77.2 | 66.3 | 71.4 |
| 5 | 78.6 | 71.9 | 75.1 |
| 6 | 80.5 | 74.4 | 77.3 |
| 7 | 82.0 | 78.5 | 80.2 |
| 8 | 84.2 | 80.1 | 82.1 |
| 9 | 85.8 | 82.5 | 84.1 |
| 10 | 87.7 | 83.9 | 85.8 |
| 11 | 88.7 | 86.5 | 87.6 |

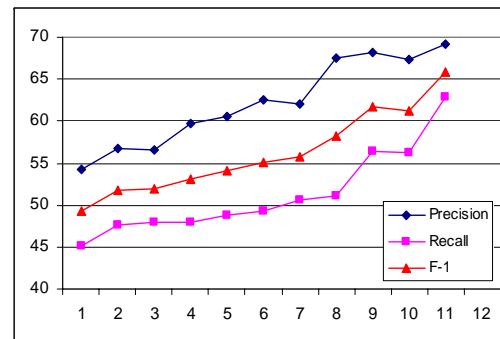
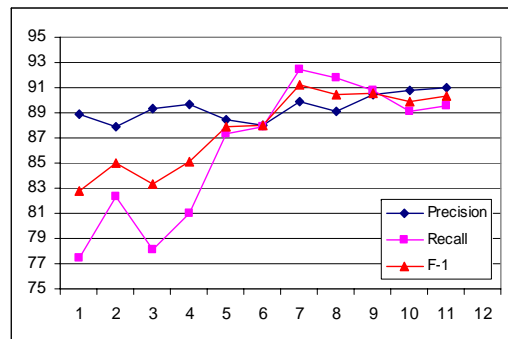


Discussion I: recognition quality

Quality curves for overall recognition



Quality curves for recognition of annotated and unannotated NIL terms



Discussion II: annotation efficiency

- Efficiency is improved by 85.0% in annotating the last 1112 NIL text pieces
- Annotation time (minutes) used in 12 annotation rounds

| Round. No. | Annotated NIL Exp. | Unannotated NIL Exp. | Total minutes |
|------------|--------------------|----------------------|---------------|
| 0 | 0 | 5031 | 5031 |
| 1 | 184 | 1580 | 1764 |
| 2 | 221 | 1431 | 1652 |
| 3 | 264 | 1236 | 1500 |
| 4 | 253 | 1236 | 1489 |
| 5 | 272 | 1070 | 1342 |
| 6 | 319 | 925 | 1244 |
| 7 | 332 | 876 | 1208 |
| 8 | 326 | 770 | 1096 |
| 9 | 348 | 705 | 1053 |
| 10 | 390 | 455 | 845 |
| 11 | 449 | 391 | 840 |



Error analysis for the automated annotation tool

- Err.1 Ambiguous NIL terms
 - “答谢粉丝(da2 xie4 fen3 si1, thank the fans)”
 - “今天吃粉丝(jin1 tian1 chi1 fen3 si1, eat vermicelli today)”
 - ***Forty*** errors with this type happened in our experiment round 11
- Err.2 Unannotated NIL terms
 - “盒饭(he2 fan4, fans of He Jie)” (He Jie is a Chinese girl.)
 - ***Five*** errors with this type happened in our experiment round 11



Conclusions

- With increasing volume of annotated NIL text pieces, quality of automated annotation of incoming NIL text pieces can be improved gradually to around 88.7% in terms of precision.
- Annotation efficiency can be improved by 85.0 with the two-stage incremental annotation approach.



Future works

- We will make use of normal corpora, say PKU corpus, to improve quality of NIL text processing technologies based on NIL corpus (which is small).



Thank you for your attention.
Questions please.

