Full citation: Min, K., MacDonell, S.G., & Moon, Y.-J. (2006) Heuristic and rule-based knowledge acquisition: classification of numeral strings in text, in Proceedings of the 2006 Pacific Rim Knowledge Acquisition Workshop (PKAW). Guilin, China, Springer (Lecture Notes in Artificial Intelligence v.4303), pp.40-50.
http://dx.doi.org/10.1007/11961239_4

Heuristic and Rule-based Knowledge Acquisition: Classification of Numeral strings in Text

Kyongho Min
School of Computer & Info Sciences
Auckland University of Technology
Private Bag 92006, Auckland 1142,
New Zealand
kyongho.min@aut.ac.nz

Stephen MacDonell
SERL
Auckland University of Technology
Private Bag 92006, Auckland 1142,
New Zealand
stephen.macdonell@aut.ac.nz

Yoo-Jin Moon
Department of MIS
Hankook University of Foreign Studies, Korea
yjmoon@hufs.ac.kr

Abstract

This paper describes the rule-based classification of numerals and strings that include numerals, composed of a number and semantic unit(s) that indicate a SPEED, NUMBER, or other measure, at three levels: morphological, syntactic, and semantic. The approach employs three interpretation processes: word trigram construction with tokeniser, rule-based processing of number strings, and n-gram based classification. We extracted numeral strings from 378 online newspaper articles, finding that, on average, they comprised about 2.2% of the words in the articles. To manually extract n-gram rules to disambiguate the number strings' meanings, our approach was trained on 886 numeral strings and tested on the remaining 3251 strings.. We implemented two heuristic disambiguation methods based on each category's frequency statistics collected from the sample data, and precision ratios of both methods were 86.8% and 86.3% respectively. This paper focuses on the acquisition and performance of different types of rules applied to numeral strings classification.

1. INTRODUCTION

Most efforts directed towards understanding natural language in text focus on sequences of alphabetical character strings. However, the text may include different types of data such as numeric (e.g. "25 players") - or alpha-numeric (e.g. "25km/h") - with/without special symbols (e.g. "$2.5 million") [5]. In current natural language processing (NLP) systems, such strings are treated as either a numeral (e.g. "25 players") or as a named entity (NE e.g. "$2.5 million") at the lexical level. However, ambiguity of semantic/syntactic interpretation can arise for such strings at the lexical level only: for example, the number "21" in the phrase "he turns 21 today" can on the surface be interpreted as any of the following: (a) as a numeral of NP (noun phrase) – indicating NUMBER; (b) as a numeral of NP – indicating the DAY of a date expression; or (c) as a numeral of NP – indicating AGE at the lexical meaning level. This type of numeral string is called a *separate numeral string* (e.g. the quantity in "survey of *801* voters") in this paper. Some numeral strings would not be ambiguous because of their meaningful units, and they are referred to as *affixed numeral strings* (e.g. speed in "his serve of *240km/h*").

In the case of separate numeral strings, some structural patterns (e.g. DATE) or syntactic functional relationships (e.g. QUANTITY as either a modifier or a head noun) could be useful in their interpretation. However, affixed numeral strings require the understanding of some meaningful units such as SPEED ("km/h" in "250km/h"), LENGTH ("m" in "a 10m yacht"), and DAY_TIME ("am", "pm" in "9:30pm").

Past research has rarely studied the understanding of varieties of numeral strings. Semantic categories have been used for named entity recognition (e.g. date, time, money, percent etc.) [7] and for a Chinese semantic classification system [13]. Semantic tags (e.g. date, money, percent, and time) and a character tokeniser to identify semantic units [1] were applied to interpret limited types of numeral strings. Numeral classifiers to interpret money and temperature in Japanese [11] have also been studied. The ICE-GB grammar [8] treated numerals as one of cardinal, ordinal, fraction, hyphenated, multiplier with two number features - singular and plural.

Polanyi and van den Berg [9] studied anaphoric resolution of quantifiers and cardinals and employed quantifier logic framework. Zhou and Su [14] employed an HMM-based chunk tagger to recognise and classify names, times, and numerical quantities with 11 surface

sub-features and 4 semantic features like FourDigitNum (e.g. 1990) as a year form, and SuffixTime (e.g. a.m.) as a time suffix (see also [3] and [10] for time phrases in weather forecasts). FACILE [2] in MUC used a rule-based named entity recognition system incorporating a chart-parsing technique and semantic categories such as PERSON, ORGANISATION, DATE, and TIME.

We have implemented a numeral interpretation system that incorporates word trigram construction using a tokeniser, rule-based processing of number strings, and n-gram based disambiguation of classification (e.g. a word trigram - left and right strings of a numeral string). The rule-based number processing system analyses each number string morpho-syntactically in terms of its type. In the case of a separate numeral string, its assumed categories are produced at the lexical level. For example, "20" would be QUANT, DAY, or NUMBER at the lexical level. However, affixed numeral strings require rule-based processing based on morphological analysis because the string has its own meaningful semantic affixes (e.g. speed unit in "24km/h"). In this paper, the different types of rule needed to classify numeral strings are described in detail.

In the next section, the categories and rules used in this system are described. In section 3, we describe the understanding process for both separate and affixed numeral strings in more detail, and focus on classification rules. Section 4 describes preliminary experimental results obtained with this approach, and discussion and conclusions follow.

2. SYNTACTIC-SEMANTIC CATEGORIES AND RULES

In this section, semantic and syntactic categories and rules (i.e. context-free rules used for affixed numeral strings) used to parse numeral strings in real text are described.

This system uses both syntactic and semantic categories to understand separate and affixed numeral strings, because a numeral string such as "20" (i.e. separate numeral string) can be understood by itself (as in "20 pages") or with reference to a structural relationship to adjacent strings (as in "on September 20 2003"). The separate numeral string "20" in "20 pages" can be interpreted as a QUANTITY to modify the noun "pages". However, knowledge of the specific DATE representation (structural relationships between adjacent strings) in "on September 20 2003" is needed to understand "20" as DAY. This requirement is even more evident with "7/12/2003" which can mean July 12, 2003 (US) or 7 December, 2003 (e.g. in Australia and New Zealand). Thus semantic categories including DAY, MONTH, and YEAR are used for date representation.

We use 40 syntactic and semantic categories, including specific semantic categories for some numeral strings (e.g. semantic categories (e.g. MONEY, DATE) and syntactic categories – (e.g. NUMBER, FLOATNUMBER, FMNUMBER) – Table 1). For example, the category FMNUMBER (ForMatted Number) signals numbers that frequently include commas every 3 digits to the left of the unit digit for ease of reading, as in "*5,000* peacekeepers."

There are two types of dictionaries in our system: one for normal English words with syntactic information such as lexical category, number, and verb's inflectional form. The other dictionary (called the user-defined dictionary) includes symbol tokens (e.g. "(", ")") and units (e.g. "km", "m"). For example, the lexical information for "km" is (:POS (Part of Speech) LU (Length Unit)) with its meaning KILOMETER.

Table 1. Sample categories and their examples

Category	Example in Real World Text
Age	"mature *20-year-old* contender", "he turns *21* today"
Date	"*20.08.2003*"
Day	"August *11* 2005"
Daytime	"between *9:30am* and *2am*", "at *3* o'clock"
Floatnumber	"support at *26.8* per cent"
FMnumber	"took command of *5,000* peacekeepers"
Length	"a *10m* yacht"
Money	"spend *US$1.4* billion"
Name	"Brent crude *LCOc1*"
Number	"*8000* of the Asian plants"
Ordinal	"a cake for her *18th* birthday"
Plural	"putting a *43-man* squad"
Phone-Number	"ph: (09) 917 1234"
Quant	"survey of *801* voters"
Range	"for *20-30* minutes"
Scores	"a narrow *3-6* away loss to Otago"
Speed	"His serve of *240km/h* this season"
Street-Number	"Address: *123* Moutain rd Mt. Eden"
Temperature	"temperatures still above *40C*"
Year	"by September *2026*"

The system uses 64 context-free rules to represent the structural form of affixed numeral strings. Each rule describes relationships between syntactic/semantic categories of the components (e.g. a character or a few characters and a number) produced by morphological analysis of the affixed string. Each rule is composed of a LHS (left hand side), RHS (right hand side), and constraints on the RHS (e.g. DATE → (DAY DOT MONTH DOT YEAR), Constraints: ((LEAPDATEP DAY MONTH YEAR))). The interpretation rules are discussed in the next section in detail.

3. NUMERAL STRING CLASSIFICATION

The numeral string interpretation algorithm is composed of three processes: a morphological analysis module, a rule-based interpretation module, called ENUMS (English NUMber understanding System), which employs both a CFG (Context-Free Grammar) augmented by constraints and a parser, and a category disambiguation module to select the best category of an ambiguous numeral string by using word trigrams.

3.1 Morphological Analysis of Numeral Strings

Affixed numeral strings such as "240km/h serve" and "a 10m yacht" require knowledge of their expression formats (e.g. speed → number + distance-unit + slash + time-unit) for understanding. For example, the string "240km/h" is analysed morphologically into "240" + "km" + "/" + "h". Our morphological analyser considers embedded punctuation and special symbols. In the case of the string "45-year-old", the morphological analyser separates it into "45" + "-" + "year" + "-" + "old". Thus we use the term, morphological analysis, rather than tokenisation because each analysed symbol is meaningful in numeral string interpretation. Table 2 shows some more results from the morphological analyser.

Table 2. Examples of morphological analysis of numeral strings

Category	Example	Morphological Analysis
MONEY	"($12.56)"	"(" + "$" + "12" + "." + "56" + ")"
DATE	"20.08.2003"	"20" + "." + "08" + "." + "2003"
FMNUMBER	"2,000"	"2" + "," + "000"
SPEED	"240km/h"	"240" + "km" + "/" + "h"
RANGE/SCORES	"20-30"	"20" + "-" + "30"
DAYTIME	"9:30am"	"9" + ":" + "30" + "am"
FLOATNUMBER	"0.03"	"0" + "." + "03"
CAPACITY	"8.2µmol/L"	"8.2" ("8" + "." + "2") + "µmol" + "/" + "L"
PLURAL	"1980s"	"1980" + "s"

After analysing the string, dictionary lookup and a rule-based numeral processing system based on a simple bottom-up chart parsing technique [6] are invoked. Instances that include some special forms of number (e.g. "03" in a time, day), are not stored in the lexicon. Thus if the substring is composed of all digits, then the substring is assigned to several possible numeric lexical categories. For example, if a numeral string "03" is encountered, then the string is assigned to SECOND, MINUTE, HOUR, DAY, MONTH, and BLDNUMBER (signifying digits after a decimal point, e.g. "0.03"). If the numeral string is "13" or higher, then the category cannot be MONTH. Similar rules can be applied to DAY and other categories. However, "13" can clearly be used as a quantifier.

Non-numeral strings are processed by dictionary lookup as mentioned above, and their lexical categories used are necessarily more semantic than in regular parsing. For example, the string "m" has three lexical categories: LU (Length Unit) as a METER (e.g. "a 10m yacht"), MILLION (e.g. "$1.5m"), and TU (Time Unit) as a MINUTE (e.g. "12m 10s" - 12 minutes and 10 seconds).

After morphological processing of substrings, an agenda-based simple bottom-up chart parsing process is applied with 64 context-free rules that are augmented by constraints. If a rule has a constraint, then the constraint is applied when an (inactive) phrasal constituent is created. For example, the rule to process a date of the form "28.03.2003" is DATE → (DAY DOT MONTH DOT YEAR) with the constraint (LEAPYEARP DAY MONTH YEAR), which checks whether the date is valid. An inactive phrasal constituent DATE1 with its RHS, (DAY1 DOT1 MONTH1 DOT2 YEAR1), would be produced and the constraint applied to verify the well-formedness of the inactive constituent.

The well-formedness of DATE (e.g. "08.12.2003") is verified by evaluating the constraint (LEAPDATEP DAY MONTH YEAR). Some other rules for affixed/separate numeral string interpretation are:

RULE5 **LHS**: AGE
RHS: (NUMBER HYPHEN NOUN HYPHEN AGETAG) – e.g. "*38-year-old* man"
Constraints: ((INTEGER-NUMBER-P NUMBER) (SEMANTIC-AGE-P NOUN) (SINGULAR-NOUN-P NOUN))

RULE21 **LHS**: TEMPERATURE
RHS: (NUMBER CELC) – e.g. "*40C*"
Constraints: (INTEGER-NUMBER-P NUMBER)
Where CELC means CELsius-C

RULE22 **LHS**: RANGE
RHS: (NUMBER HYPHEN NUMBER) – e.g. "*20-30* minutes"
Constraints: (RANGE-P NUMBER NUMBER)

RULE30 **LHS**: FLOATNUMBER
RHS: (NUMBER DP BLDNUMBER) – e.g. "*20.54* percent"
Constraints: (FLOATNUMBER-P NUMBER DP BLDNUMBER)
where DP means Decimal Point and BLDNUMBER means BeLow-Decimal NUMBER.

RULE42 **LHS**: WEIGHT
RHS: (NUMBER WU) – e.g. "55kg"
Constraints: NIL.
where WU means Weight Unit.

3.2 Classification based on Word Trigrams

For separate numeral strings, the interpreted categories can be ambiguous because there is no semantic unit attached. For example, "240km/h" would be uniquely interpreted as SPEED. However, the numeral string "20" could be either QUANT (e.g. "20 boys") or DAY (e.g.

"20 May 2005") without using different context information. Thus word trigrams are used to disambiguate the syntactic/semantic categories of numeral strings.

Word trigrams are collected when a document is read and tokenised. While tokenising a string (tokenisation based on a single whitespace), the numeral string is identified with its word trigram (left and right string of the numeral string). For example, the numeral string "100" in "The company counts more than 100 million registered users worldwide." has its word trigram ("than" - left wordgram, "million" - right wordgram). If a numeral string occurs at either the start or end of a sentence, then either a left or right wordgram would be empty (i.e. NULL).

Table 3. Examples of feature types

Feature Type	Examples
Lexical Cateogry	Preposition-p: (preposition-p left wordgram ("to")) in "home to 22 superyachts"
Number information	Plural-noun-p: (plural-noun-p right wordgram ("voters")) in "801 voters"
Validity of value	Valid-day-p: (valid-day-p numeral string ("28") right wordgram ("February")) in "28 February 2004" – not "30 February 2004"
Conceptual type	Month-string-p: (month-string-p right wordgram ("February")) in "28 February 2004"
Case of a word	Capital-letter-p: (capital-letter-p left wordgram ("Lee,")) in "Lee, 41, has…"
Punctuation marks	Comma-p: (comma-p left wordgram ("Lee,")) in "Lee, 41, has…"

Disambiguation of categories is based on rules manually encoded by using sample data and each rule is based on morpho-syntactic features of the word trigrams. For example, a punctuation mark like comma is important for disambiguating the AGE category (e.g. "41" in "Lee, 41, has").

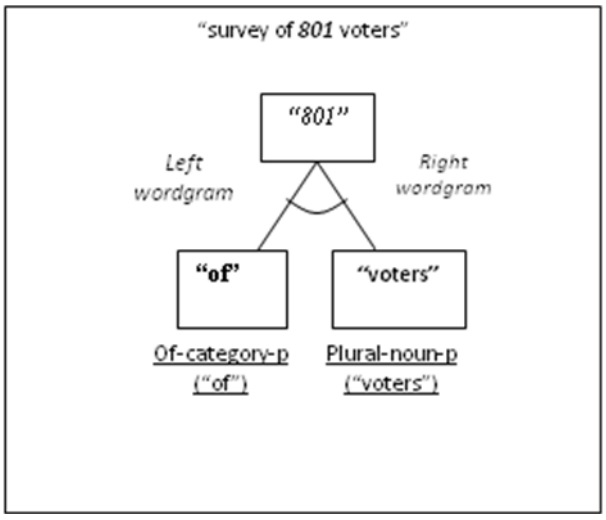

Fig. 1. Contextual constraints based on word trigrams

The features used for selection of the best meaning of an interpreted numeral string are based on syntactic and surface features. These features are joined together to reflect contextual information (e.g. using neighbour words information). The current system uses left and right adjacent words of the numeral string with the following features (Table 3): lexical category (e.g. NOUN, VERB), number information (e.g. PLURAL, SINGULAR), validity of values (e.g. valid DAY), semantic information (e.g. MONTH concept), Case of a letter (e.g. capitalisation), and punctuation marks (e.g. PERIOD, COMMA).

With the features extracted from a numeral string's word trigram, the contextual features of the word trigram are used for the selection of the 'best' category for that numeral string. The contextual information is extracted manually and its form is based on conjunction of the word trigrams features (Fig. 1).

For QUANT category disambiguation, 22 constraints are used. The word trigram (wordgram) for a numeral string "801" in the substring "survey of 801 voters" would be ("of" "801" "voters"). Thus one QUANT selection rule would be:

(and (of-category-p left-wordgram ("of"))
(plural-noun-p right-wordgram ("voters"))).

If the numeral string "20" is in the string "March 20 2003", then the category would be DAY and one of four selection rules would be:

(and (month-string-p left-wordgram ("March"))
(valid-day-p "March" "20") - not "March 35"
(number-p right-wordgram ("2003"))).

If a numeral string (e.g. "41" in "Lee, 41, has") satisfies one of three AGE rules, then the numeral string is disambiguated as AGE. The rule is:

(and (capital-letter-p left-wordgram("Lee"))
(comma-p left-wordgram (",” in "Lee,"))
(comma-p numeral-string (",” in "41,")).

This rule means that if the word in the left wordgram begins with a capital letter, if the word in the left wordgram has a comma, and if the numeral string has a comma, then the rule applies.

For DAYTIME category (e.g. "on Tuesday (*0030* NZ time Wednesday)"), semantic meanings of word trigrams are used with the numeral string's surface pattern. The rule is:

(and (weekday-string-p left-wordgram)
(= 4 (length numeral-string))
(daytime-string-p numeral-string)
(country-name-p right-wordgram)).

For YEAR category (e.g. "end by September *2026*"), heuristic constraints are used as follows:

(and (>= numeral-string 1000)
(<= numeral-string 2200)).

The contextual information based on word trigrams is applied to disambiguate multiple categories resulting from the numeral string interpretation process. Two

heuristic methods are implemented to compare their results:

- Heuristic Method 1 (Method-1) – Method 1 applies wordgram constraints and collects and then considers all satisfied constraints. For example, if the QUANT and NUMBER constraints for a numeral string are satisfied, then the two categories are used for the numeral's disambiguation. With collected categories, the annotation frequency of the categories collected from sample data (i.e. Sample data in Table 4) is used to select the best category. If the frequency of QUANT is greater than that of NUMBER in annotation statistics, then QUANT is selected for the category of the numeral string from the meanings processed by a numeral string interpretation system. If a constraint for AGE category is satisfied, then a new category, AGE, is produced because the numeral string interpretation system could not produce the category. If there is no category that satisfies the constraints, then preference rules with ordered categories based on frequency of annotation (e.g. QUANT > MONEY > DATE > etc.) are applied to select the best category.
- Heuristic Method 2 (Method-2) – Method 2 is similar to Method-1 except for the application of annotation statistics. To select the best category for an ambiguous numeral string, the rareness of annotation statistics is applied. If both YEAR and QUANT constraints are satisfied, and the annotation frequency of YEAR is less than that of QUANT, then YEAR is selected for the category of the numeral string. If there is no category that satisfies the constraints, then preference rules with ordered categories based on rareness of annotation statistics (e.g. DATE < MONEY < YEAR < etc.) are applied to select a category.

4. EXPERIMENTAL RESULTS

We implemented our system in Allegro common lisp with IDE. We collected 9 sets of online newspaper articles and used 91 articles (sample data) to build disambiguation rules for the categories of numeral strings. The remaining 287 articles (test data) were used to test the system. Among the 48498 words in the 91 sets of sample data, 886 numeral strings (1.8% of total strings and 10 numeral strings out of 533 strings for each article on average) were found. In the case of the test data, 3251 out of 144030 words (2.2% of total strings and 11 numeral strings of 502 strings for each article on average) were identified as numeral strings (Table 4).

Table 4. Data size and proportion of numeral strings

Date Name	Total Articles	Total Strings	Total Numerals	Remark
Sample	91	48498	886 (1.8%)	2 sets
Test	287	144030	3251 (2.3%)	7 sets
Total	378	192528	4137 (2.1%)	-

The proportion of numeral strings belonging to each category in both sample and test data were QUANT (826 of 3251, 20.0%, e.g. "survey of *801* voters"), MONEY (727, 17.6%, e.g. "$15m", "$2.55"), DATE (380, 9.2%, e.g. "02.12.2003"), YEAR (378, 9.1%, e.g. "in 2003"), NUMBER (300, 7.3%, e.g. "300 of the Asian plants"), SCORES (224, 5.4%, e.g. "won 25 - 11"), FLOATNUMBER (8.0%, e.g. "12.5 per cent"), and others in order.

Table 5 shows the recall/precision/F-measure ratios (balanced F-measurement) based on the two disambiguation methods. Method-2 for the test data shows better recall ratio (77.6%) than Method-1. Method-1 for the test data shows better precision ratio (86.8%) than Method-2. However, the difference between Method-1 and Method-2 is 0.5% in recall ratio and 0.5% in precision ratio, indicating that the performance of each method is close to identical.

Table 5. Recall/Precision/F-measurement ratios of two heuristic methods by data set

DateName	Recall Ratio (%)		Precision Ratio (%)		F-measure (%)	
	Method-1	Method-2	Method-1	Method-2	Method-1	Method-2
Sample	86.0	86.6	95.3	93.4	90.2	89.8
Test	77.1	77.6	86.8	86.3	81.7	81.7
Average	81.5	82.1	91.1	89.8	85.9	85.8

Table 6 shows the recall/precision/F-measure ratios, of selected categories, based on the two disambiguation methods. The average performance difference between the methods is 0.7% in recall ratio, 0.6% in precision ratio, and 0.1% in F-measurement. The results for affixed numeral strings (e.g. FLOATNUMBER, FMNUMBER, MONEY, RANGE, SCORES) showed large differences in performance (10.3% to 100%) as did the results for separate numeral strings (e.g. AGE, DAY, NUMBER, QUANT, YEAR) (38.2% to 97.5%). The performance of the RANGE category is poor because of numeral strings such as "$US5m-$US7m" and "10am-8pm". These numeral strings are presently not covered by our CFG (Context-Free Grammar).

Table 6. Recall/Precision/F-measurement ratios of two heuristic methods by categories

Category	Recall Ratio (%)		Precision Ratio (%)		F-measure (%)	
	Method-1	Method-2	Method-1	Method-2	Method-1	Method-2
Age	43.8	43.8	97.5	97.5	60.5	60.5
Day	68.7	69.3	86.8	81.9	76.7	75.1
Number	82.1	91.7	42.4	38.2	55.9	53.9
Quant	83.2	81.3	88.2	97.5	85.6	88.7

Category	Recall Ratio (%)		Precision Ratio (%)		F-measure (%)	
	Method-1	Method-2	Method-1	Method-2	Method-1	Method-2
Year	86.9	70.0	92.6	87.9	89.6	77.9
Floatnumber	98.6	98.6	96.9	96.9	97.8	97.8
Fmnumber	100	100	98.8	98.8	99.4	99.4
Money	99.2	99.2	99.9	99.9	99.5	99.5
Range	10.3	55.2	66.7	35.6	17.9	43.2
Scores	84.8	74.1	89.2	97.1	87.0	84.1
*Average	74.9	75.6	90.0	89.4	81.8	81.9

(*average is the average of all categories)

Compared to other separate categories, the system interprets the QUANT category better than YEAR and NUMBER because the disambiguation module for QUANT category has more constraints based on wordgram information (i.e. 22 constraints for QUANT, 7 for YEAR, and 6 for NUMBER).

For disambiguation process using Method-1 and Method-2, 2213 (53.5%) of 4137 numeral strings were ambiguous after numeral string interpretation process. Among these, the numbers of satisfied constraints are no constraint (746 – 33.7%), one constraint (1271 – 57.4%), 2 constraints (185 – 8.4%), and three constraints (11 – 0.5%).

5. DISCUSSION AND CONCLUSIONS

It is not easy to compare our system to other NE (Named Entity) recognition systems directly because the target recognition of named entities is different. Other systems in MUC-7 [2] and CoNLL2003 [4] focused on the general recognition task of named entities including person, location, date, money, and organisation. The systems in MUC-7 and CoNLL2003 were trained and tuned by using the necessary training corpus with document preprocessing (e.g. tagging and machine learning). However, our system is focused on understanding the varieties of numeral strings more deeply and had no training phase. The manually annotated data from sample data (25%) was used for implementation of disambiguation rules. Performance of MUC-7 systems was greater than 90% in precision. Our system correctly interpreted 88.7% of numeral strings.

Further rules and lexical information are required to process more numerals in real world text (see [5]). For better disambiguation, more fine-grained disambiguation modules based on word trigrams would be required. Currently, syntactic categories of both left and right wordgrams of each numeral string are used. The major problem of the use of syntactic category is lexical ambiguity (e.g. “in” is lexically ambiguous as preposition, adverb, and noun). To reduce the ambiguities, more fine-grained surface patterns would be required. In addition, the extension of this system to other data such as biomedical corpora [12] is required to test the overall effectiveness of our approach.

Another research avenue would be the automatic acquisition of constraints for the disambiguation process and the determination of the significance of each feature for the disambiguation process. For example, for the QUANT category, the plural number information of a right wordgram could be more significant than various information in a left wordgram. In addition, the significance of offset (adjacency) of wordgrams to extract more contextual knowledge could be studied for better disambiguation precision for future development.

In conclusion, separate and affixed numeral strings are frequently used in real text. However, there seems to be no system that interprets numeral strings systematically; they are frequently treated as either numerals or nominal entities. In this paper, we have analysed the numeral strings at lexical, syntactic, and semantic levels with some contextual information. The system is composed of a tokeniser with word trigram constructor, numeral string processor which includes a morphological analyser and a simple bottom-up chart parser with context-free rules augmented by constraints, and a disambiguation module based on word trigrams. The numeral string interpretation system successfully interpreted 88.7% of test data. The system could be scaled up to cover more numeral strings by extending the lexicon and rules.

REFERENCES

1. Asahara, M., Matsumoto Y.: Japanese Named Entity Extraction with Redundant Morphological Analysis. Proceedings of HLT-NAACL 2003. (2003) 8-15
2. Black, W., Rinaldi, F., Mowatt, D.: FACILE: Description of the NE system used for MUC-7. Proceedings of MUC-7. (1998)
3. Chieu, L., Ng, T.: Named Entity Recognition: A Maximum Entropy Approach Using Global Information. Proceedings of the 19th COLING. (2002) 190-196
4. CoNLL-2003 Language-Independent Named Entity Recognition. http://www.cnts.uia.ac.be/conll2003/ner/2. (2003)
5. Dale, R.: A Framework for Complex Tokenisation and its Application to Newspaper Text. Proceedings of the second Australian Document Computing Symposium. (1997)
6. Earley, J.: An Efficient Context-Free Parsing Algorithm. CACM. 13(2) (1970) 94-102
7. Maynard, D., Tablan, V., Ursu, C., Cunningham, H., Wilks, Y.: Named Entity Recognition from Diverse Text Types. Proceedings of Recent Advances in NLP. (2001)
8. Nelson, G., Wallis, S., Aarts, B.: Exlporing Natural Language - working with the British Component of the International Corpus of English, John Benjamins, The Netherlands. (2002)
9. Polanyi, L., van den Berg, M.: Logical Structure and Discourse Anaphora Resolution. Proceedings of ACL99 Workshop on The Relation of Discourse/Dialogue Structure and Reference. (1999) 10-117
10. Reiter E., Sripada, S.: Learning the Meaning and Usage of Time Phrases from a parallel Text-Data

Corpus. Proceedings of HLT-NAACL2003 Workshop on Learning Word Meaning from Non-Linguistic Data. (2003) 78-85
11. Siegel, M., Bender, E. M.: Efficient Deep Processing of Japanese. Proceedings of the 3rd Workshop on Asian Language Resources and International Standardization. (2002)
12. Torii, M., Kamboj, S., Vijay-Shanker, K.: An investigation of Various Information Sources for Classifying Biological Names. Proceedings of ACL2003 Workshop on Natural Language Processing in Biomedicine. (2003) 113-120
13. Wang, H., Yu, S.: The Semantic Knowledge-base of Contemporary Chinese and its Apllication in WSD. Proceedings of the Second SIGHAN Workshop on Chinese Language Processing. (2003) 112-118
14. Zhou, G., and Su, J. Named Entity Recognition using an HMM-based Chunk Tagger. Proceedings of ACL2002. (2002) 473-480