A Preliminary Work on Classifying Time Granularities of Temporal Questions

Wei Li[1], Wenjie Li[1], Qin Lu[1], and Kam-Fai Wong[2]

[1] Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Hong Kong
{cswli, cswjli, csluqin}@comp.polyu.edu.hk
[2] Department of Systems Engineering, the Chinese University of Hong Kong, Shatin, Hong Kong
kfwong@se.cuhk.edu.hk

Abstract. Temporal question classification assigns time granularities to temporal questions ac-cording to their anticipated answers. It is very important for answer extraction and verification in the literature of temporal question answering. Other than simply distinguishing between "date" and "period", a more fine-grained classification hierarchy scaling down from "millions of years" to "second" is proposed in this paper. Based on it, a SNoW-based classifier, combining user preference, word N-grams, granularity of time expressions, special patterns as well as event types, is built to choose appropriate time granularities for the ambiguous temporal questions, such as When- and How long-like questions. Evaluation on 194 such questions achieves 83.5% accuracy, almost close to manually tagging accuracy 86.2%. Experiments reveal that user preferences make significant contributions to time granularity classification.

1 Introduction

Temporal questions, such as the questions with the interrogatives “when”, “how long” and “which year”, seek for the occurrence time of the events or the temporal attributes of the entities. Temporal question classification plays an important role in the literature of question answering and temporal information processing. In the evaluation of TREC 10 Question-Answering (QA) track [1], more than 10% of questions in the test question corpus are temporal questions. Different from TREC QA track, Workshop TERQAS (*http://www.timeml.org/terqas/*) particularly investigated on temporal question answering instead of a general one. It focused on temporal and event recognition in question answering systems and paid great attention to temporal relations among states, events and time expressions in temporal questions. TimeML (*http://www.timeml.org*), a temporal information (e.g. time expression, tense & aspect) annotation standard, has also been used for temporal question answering in this workshop [2]. Correct understanding of a temporal question will greatly help extracting and verifying its answers and certainly improve the performance of any question answering system. Look at the following examples.

[Ea]. What is the birthday of Abraham Lincoln?
[Eb]. When did the Neanderthal man live?

R. Dale et al. (Eds.): IJCNLP 2005, LNAI 3651, pp. 414 – 425, 2005.
© Springer-Verlag Berlin Heidelberg 2005

In a general question answering system, the question classifier commonly classifies temporal questions into two classes, i.e. "date" and "period". With such a system, the above two questions are both assigned a "date". Whereas it is natural for the question [Ea] to be answered with a particular data (e.g. "12/02/1809"), it is not the case for question [Eb], because a proper answer could be "35,000 years ago". However, if it is known that the time granularity concerned is "thousands of years", answer extraction turn to be more targeted. The need for a more fine-grained classification is obvious. Although there were different question classification hierarchies, as reported [3,4,12,13,14], few inclined to introducing the classification hierarchy (e.g. "year", "month" and "day") which could give a clearer direction to guide answer extraction and verification of temporal questions. In the following, we try to find out whether temporal questions can be further classified into finer time granularity and how to classify them.

By examining a temporal question corpus consisting of 348 questions, 293 of which are gathered from UIUC question answering labelled data (*http://l2r.cs.uiuc.edu/~cogcomp/Data/QA/QC*), and the rest 55 from TREC 10 test corpus, we find two different cases. On the one hand, some questions are very straightforward in expressing the time granularities of the answers expected, e.g. the questions beginning with "which year" or "for how many years". On the other hand, some questions are not so obvious, e.g. the questions headed by "when" or "for how long". We call such questions ambiguous questions. Not surprisingly, the ambiguous *When-* and *How long*-like questions account for a large proportion in this temporal question corpus, i.e. 197 from 348 in total.

We further investigate on those 197 ambiguous questions in order to find out whether they can be classified into finer time granularity. Three experimenters are requested to tag a time granularity to each question independently[1]. Answers are not provided. The tag with two agreements is taken as the time granularity class of the corresponding temporal question. Otherwise the tag "UNKNOWN" is assigned. Reference answers for the questions are extracted from AltaVista Web Search *(http://www.altavista.com)*. Comparing the time granularities tagged manually with those provided by the reference answers, we find that only 27 out of 197 questions are incorrectly tagged, in other words, the manually tagging accuracy is 86.2%. Errors exist though, the relatively high agreement between users' tagging and reference answers lights the hope of automatically determining the time granularities of temporal questions.

Analysing the tagging results, it is revealed that the tagging errors arouse from three sources: insufficient world knowledge, different speaking habits and different expected information granularity among human. See the following examples:

[Ec]. When did the Neanderthal man live?
User: year; Ref.: thousands of years
[Ed]. How long is human gestation?
User: month; Ref.: week
[Ee]. When was the first Wall Street Journal published?
User: year; Ref.: day

[1] The granularity hierarchy and the tagging principle will be detailed later.

For question [Ec], the time granularity should be "thousands of years", rather than "year". This error could be corrected if one knows that Neanderthal man existed 35,000 years ago. The time granularity of question [Ed] should be "week", but not "month" in accordance with the habit. For question [Ee], users' tag is "year", different from the reference answer's tag "day". However, both granularities are acceptable in commonsense, because the different users may want coarser or finer information. This observation suggests that incorporating question context, world knowledge, and speaking habits would help determine the time granularities of temporal questions.

In this paper, we propose a fine-grained temporal question classification scheme, i.e. time granularity hierarchy, consisting of sixteen non-exclusive classes and scaling down from "millions of years" to "second". The SNoW-based classifier is then built to combine linguistic features (including word *N*-grams, granularity of time expressions and special patterns), user preferences and event types, and assign one of the sixteen classes to each temporal question. In our work, user preference, which characterizes world knowledge and speaking habits, is estimated by means of the time granularities of the entities and/or events involved. The SNoW-based classifier achieves 83.5% accuracy, almost close to 86.2% of manually tagging accuracy. Experiments also show that user preference makes a great contribution to time granularity classification.

The rest of this paper is organized as follows. In the next section various related works in this literature are introduced. In Sect. 3, we demonstrate the time granularity hierarchy and principles. User preference is fully investigated in Sect. 4. Feature design is depicted in Sect. 5. Time granularity classifiers are introduced in Sect. 6 and the experiment results are presented in Sect. 7. We finally conclude this paper in the last section.

2 Related Works

In TREC QA track, almost every QA system joining in the evaluation has a question classification module. This makes question classification a hot topic. Questions can be classified from several aspects. Most classification hierarchies [3,4,12,13,14] adopt the anticipated answer types as its classification criteria. Abney et al. [4] gave a coarse classification hierarchy with seven classes (person, location, etc.). Hovy et al. [13] introduced a finer classification with forty-seven classes manually constructed from 17,000 practical questions. Li et al. [3] proposed a two-level classification hierarchy, a coarser one with six classes and a finer one with fifty classes. In all these classification hierarchies, temporal questions are simply classified into two classes, i.e. "date" and "period". Some works classified temporal questions from other aspects. In [2], a temporal question classification hierarchy is proposed according to the temporal relation among state, event and time expression. In [5], temporal questions are classified into three types with regard to question structure: non-temporal, simple and complex. Diaz F. et al. [6] did an interesting work on the statistics of the number of topics along timeline. According to whether questions or topics have a clear distribution along timeline, they can be classified into three types: atemporal, temporal clear and temporal ambiguous. Focusing on ambiguous temporal questions, e.g. *when* and *how long*-like questions, we introduce a classification hierarchy in terms of the anticipated answer types.

It is an extension of two classes "date" and "period" and includes sixteen non-exclusive classes scaling down from "millions of years" to "second".

Related to the work of features design, Li et al. [3] built the question classifier based on three types of features, including surface text (e.g. N-grams), syntactic features (e.g. part-of-speech and name entity tags), and semantic related words (words that often occur with a specific question class). Later works of Li et al. [10] introduced semantic information and world knowledge from external resources such as WordNet. In this paper, we introduce a new feature, user preference, which is expected to imply the world knowledge in time granularity in the experiment. User preference is estimated from statistics with which Diaz F. et al. [6] determine whether a question is temporal ambiguous or not. E. Saquete et al. [5] suggested that questions had different structures, i.e. non-temporal, simple and complex, which is helpful to handle questions more orderly. It gives us inspiration to use question focus, i.e. whether a question is event-based or entity-based.

Many machine-learning methods have been used in question classification, such as language model [7], SNoW [3,10], maximum entropy [15] and support vector machine [8,9]. In our experiments, language model is selected as the baseline model, and SNoW is selected to tackle to the large feature space and build the classifier. In fact, SNoW has already been used in many other fields, such as text categorization, word sense disambiguation and even facial feature detection.

3 Time Granularity Hierarchy and Tagging Principles

In traditional question answering systems, only two question types are time-related, i.e. "date" and "period". For the reasons explained in Sect. 1, we propose a more detailed temporal question classification scheme, namely time granularity hierarchy scaling down from "millions of years" to "second" in order to facilitate answer extraction and verification. The initial time granularity hierarchy includes the following twelve classes: "second", "minute", "hour", "day", "week", "month", "season", "year", "decade", "century", "thousands of years" and "millions of years".

Granularity "weekday" is added to the initial hierarchy because some temporal questions favor "weekday" instead of "day", although both of them indicate one day. Some questions favour a region of time granularity. Look at the following examples.

[Ef]. What time of year has the most air travel?
[Eg]. What time of day did Emperor Hirohito die?

For [Ef] question, its time granularity could be "season", "month" or even "day"; and for question [Eg], the time granularity could be "hour" or "minute". We can only determine that their time granularities are less than "year" or "day" respectively, but cannot go any further. Such situations only occur to time granularity "year" and "day", so we expand the original classification hierarchy by adding another two types: "less than day", "less than year". Besides, the questions asking for festivals are classified into "special date".

Up to now, the time granularity hierarchy has sixteen classes. The less frequent temporal measures, such as "microsecond" and "billions of years" are ignored. As mentioned above, the class "less than day" overlaps several granularities, e.g. "hour" and "minute", so the time granularity hierarchy we proposed is non-exclusive.

In reality, some temporal questions can be answered in several different time granularities. For example, question "when was Abraham Lincoln born?", its answers can be a "day" ("12/02/1809") or a "year" ("1809"). To resolve this confliction, we adopt two principles for time granularity annotation.

[Pa]. Assign the minimum time granularity we can determine to a given temporal question if several time granularities are applicable.
[Pb]. Select the time granularity with regard to speaking habits or user preferences.

When the two principles conflict to each other, principle [Pb] takes the priority. With principle [Pa], time granularity of the above question can only be "day".

4 User Preference

In general, temporal questions have two different focuses: entity-based and event-based.

[a]. Entity-based question: temporal interrogative words + (be) + entity, e.g. "When was the World War II?"
[b]. Event-based question: temporal interrogatives + event, e.g. "When did Mount St. Helen last have a significant eruption?"

Time granularities of entities (or events) have great significance to those of entity-based (or event-based) temporal questions. So, in the following, we make estimation of the time granularities of entities and events from statistics, based on the intuition that some entities or events may favor certain types of time granularities, which is called user preference here.

4.1 Estimation of Time Granularities of Entities and Events

4.1.1 Time Granularity of Entities

The time granularity of the entity is derived by counting the co-occurrences of the entity and time granularities. The statistics is gathered from AltaVista Web Search. The sentences containing both the entity and time expressions are extracted from the first one hundred results returned by AltaVista with the entity as the searching keyword. The probability P of a time granularity class tg_i on the occurrence of the entity is calculated as the following Equation (1).

$$P(tg_i \mid entity) = \frac{\#(tg_i \cap entity)}{\#(entity)} \quad TG(entity) = Arg\max_{tg_i} P(tg_i \mid entity) \tag{1}$$

#( ) is the number of the sentences containing the expressions between the parenthesis. *TG(entity)* represents the time granularity of the entity.

4.1.2 Time Granularity of Events

The time granularities of the events are not directly extracted as what is done to the entities, because they have little chance to be reused on the observation that there are rarely two identical events in a question corpus. As an alternative, the time granularity of an event is estimated from a sequence of entity-verb-entity' approximating the event. The time granularity of the verb is determined as Equation (1) by substituting

"*verb*" for "*entity*". We choose two strategies for the estimation: maximum product and one-win-all.

Maximum product: $$P(tg_i \mid event) = \frac{1}{Z} P(tg_i \mid entity) P(tg_i \mid verb) P(tg_i \mid entity')$$

$$TG(event) = Arg\max_{tg_i} P(tg_i \mid event) \tag{2}$$

TG(event) represents time granularity of event. *Z* is used for normalization.

One-win-all: $$TG(event) = Arg\max_{tg_i} \{P(tg_i \mid entity), P(tg_i \mid verb), P(tg_i \mid entity')\} \tag{3}$$

Equation (1) is smoothed in order to avoid 0 values in Equation (2).

$$P(tg_i \mid w) = \frac{\#(tg_i \cap w) + 1}{\#(w) + t} \qquad t = \left| tg_i \right| \tag{4}$$

t is the number of the time granularity classes, *w* is either an entity or a verb.

4.1.3 Experiment: Evaluating the Estimation

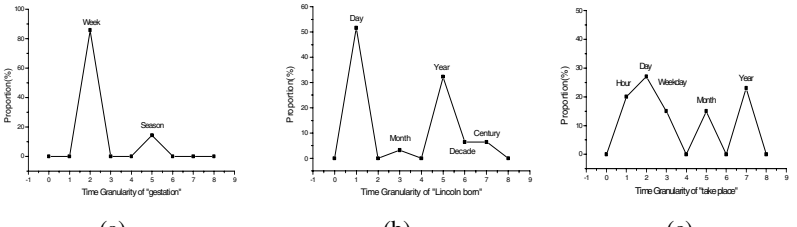
In the 197 ambiguous questions, 12 questions are entity-based, and the rest 185 questions are event-based. If all the 197 questions are arbitrarily assigned a tag "year", the tagging accuracy is 48.2%.

For each entity-based or event-based question, the time granularity of the entity or event within it are assumed as the time granularity of the question. Compared with the time granularity of the reference answer, for the entity-based questions, we achieve 75% accuracy; for the event-based question, the accuracy of maximum product strategy and one-win-all strategy are 67.0% and 64.3% respectively. It seems that maximum product strategy is more effective than one-win-all strategy in this application. With maximum product strategy, the overall accuracy on all the 197 ambiguous questions is 67.4%. Notice that the accuracy of arbitrarily tagging is only 48.2%, so the estimation of the time granularities of the entities and the events is useful for determining the time granularities of temporal questions.

4.2 Distribution of the Time Granularity of Entities and Events

4.2.1 Observation of Distribution

In the experiments of estimation, we find that some entities or events tend to favor only one certain time granularity, some others tend to favor several time granularities, and the rest may have a uniform distribution almost on every time granularity.



(a) (b) (c)

Fig. 1. Distribution of the time granularities of the entities and events

In Fig. 1(a), time granularity "day" takes a preponderant proportion, i.e. more than 80%, in the distribution of "gestation", which is called single-peak-distribution. In Fig. 1(b), both "day" and "year" take a large proportion, so "Lincoln born" is multi-peak-distributed. In Fig. 1(c), for "take place", all the time granularities almost take a similar proportion and it is a uniform distribution.

4.2.2 Experiments on Distribution

Assume an entity (or event) E, its possible time granularities $\{tg_i, i=1,\ldots t\}$ and the corresponding probabilities $\{P_i, i=1,\ldots t\}$ (calculated by Equation 1 and 2).

$$\mu = \frac{1}{t}\sum_i P_i \;;\; d = \sum_i I(P_i, \mu) \;;\; I(P_i, \mu) = \begin{cases} 1 & P_i > \mu \\ 0 & P_i \le \mu \end{cases} \tag{5}$$

d is the number of time granularities tg_i with higher probability P_i than average probability μ. For simplicity, distribution D_E of the time granularity of E is determined as follows,

$$D_E = \begin{cases} Single & d = 1 \\ Multi & 1 < d \le 3 \\ Uniform & d > 3 \end{cases} \tag{6}$$

Observing the experiment results in Sect. 4.1.3, 88.7%, 56.3% and 18.9% accuracy are achieved on the questions within which the time granularities of the entities or events are estimated to be single-peak-, multi-peak-, and uniform-distributed respectively. So whether the estimated time granularity of the entity or event is single-peak-, multi-peak-, or uniform-distributed highlights the confidence on the estimation, which can be taken as a feature associated with the estimation of the time granularities.

5 Feature Design

As described in the above section, estimation of the time granularities of the entities and the events is useful for determining the time granularities of temporal questions; whether a question is entity-based or not and the distribution of time granularities of the entities and events within the questions will also be taken as associated features. These three features are named user preference feature in total. Besides, another four types of features are considered.

Word N-grams

Word N-grams feature, e.g. unigram and bigram is the most straightforward feature and commonly used in question classification. In general question classification, unigram "when" indicates a temporal question. In temporal question classification, unigram "birthday" always implies a "day" while bigram "when … born" is a strong evidence of the time granularity "day". From this aspect, word N-grams also reflect user preference on time granularity.

Granularity of Time Expressions

Time expressions are common in temporal questions, e.g. "July 11, 1998" and date modifier "1998" in "1998 Superbowl". We take the granularities of time expressions as features, for example,

TG("in 1998") = "year" TG("July 11, 1998") = "day"

Granularities of time expressions impose the constraints on the time granularities of temporal questions. If there is a time expression whose time granularity is *tg* in a temporal question, time granularity of this question can not be *tg*. For example, question "When is the 1998 SuperBowl?", its time granularity can not be "Year", i.e. the time granularity of "1998".

Special Patterns
In word N-gram features, words are equally processed, however, some special words combining with the verbs or the temporal connectives (e.g. "when", "before" and "since") will produce special patterns and affect the time granularities of temporal questions. Look at the following examples.

[Eh]. Since when hasn't John Sununu been able to fly on government planes for personal business?
[Ei]. What time of the day does Michael Milken typically wake up?

For question [Eh], the temporal preposition "since" combined with "when" highlights that this question is seeking for a beginning point time, which implies a finer time granularity; for question [Ei], "typically" combined with verb "wake up" indicates a generally occurred event, and implies that its time granularity could be "less than day" or "less than year".

Event Types
In general, there are four event types: states, activities, accomplishments, and achievements. States and activities favour larger time granularities, while accomplishments and achievements favour smaller ones. For example, the activity "stay" will favour larger time granularity than the accomplishment event "take place".

6 Classifier Building

In this work, we choose the Sparse Network of Winnow (SNoW) model as the time granularity classifier and compare it with a commonly used Language Model (LM) classifier.

6.1 Language Model (LM)

As language model has already been used in question classification [7], it is taken as the baseline model in the experiments. Language model mainly combines two types of features, i.e. unigram and bigram. Given a temporal question *Q*, its time granularity *TG(Q)* is calculated by Equation (7).

$$TG(Q) = Arg\max_{tg_i} \lambda \prod_{j=1}^{j=m} P(tg_i \mid w_j) + (1-\lambda) \prod_{j=1}^{j=n} P(tg_i \mid w_j w_{j+1}) \quad (7)$$

w represents words. *m* and *n* are the numbers of unigrams and bigrams in questions respectively. λ assigns different weights to unigrams and bigrams. In the experiment, best accuracy is achieved when $\lambda = 0.7$ (see Sect. 7.3.1).

6.2 Sparse Network of Winnow (SNoW)

SNoW is a learning framework and applicable to the tasks with a very large number of features. It selects active features by updating weights of features, and learns a

linear function from a corpus consisting of positive and negative examples. Let $Ac=\{i_1, \ldots, i_m\}$ be the set of features that are active and linked to target class c. Let s_i be the real valued strength associated with feature in the example. Then the example's class is c if and only if,

$$\sum_{i \in Ac} w_{c,i} s_i \geq \theta_c \tag{8}$$

$w_{c,i}$ is weight of feature i connected with class c, which is learned from the training corpus. SNoW has already been used in question classification [3,10] and good results are reported. As mentioned in Sect. 5, five types of features are selected for our task. They are altogether counted to more than ten thousand features. Since it is a large feature set, SNoW is a good choice.

7 Experiments

7.1 Setup

In this 348-question-corpus (see Sect. 1), time granularities of 151 questions are straightforward, while those of the rest 197 questions are ambiguous. For the sixteen time granularity classes, we only consider ten classes including more than four questions. Questions with unconsidered time granularity classes excluded, the question corpus has 339 questions in total, 145 for training and 194 for testing. As a result, the task is to learn a model from the 145-question training corpus and classify questions in the 194-question test corpus into ten classes: "second", "minute", "hour", "day", "weekday", "week", "month", "season", "year" and "century". The SNoW classifier is downloaded from UIUC *(http://l2r.cs.uiuc.edu/~cogcomp/download.php?key=SNOW)*.

7.2 Evaluation Criteria

The primary evaluation standard is accuracy$_1$, i.e. the proportion of the correct classified questions out of the test questions (see Equation 9). However, if a question seeking for a finer time granularity, e.g. "day", has been incorrectly determined as a coarser one, e.g. "year", it should also be taken as partly correct, which is reflected in accuracy$_2$ (see Equation 10).

$$Accuracy_1 = \frac{\#(correct)}{\#(test)} \tag{9}$$

#( ) is number of questions.

$$Accuracy_2 = \frac{\sum_i RR(Q_i)}{\#(test)} \qquad RR(Q) = \begin{cases} 1 & R(tg_Q') = R(tg_Q) \\ 0 & R(tg_Q') < R(tg_Q) \\ 1/(R(tg_Q') - R(tg_Q) + 1) & R(tg_Q') > R(tg_Q) \end{cases} \tag{10}$$

tg_Q and tg_Q' are the reference and classification result respectively. $R(tg_Q)$ is the rank of the time granularity class tg_Q, scaling down from "millions of years" to "second". Rank of "second" is 1, while rank of "year" is 9. The ranks of the last three

time granularities, i.e. “special date”, “less than day” and “less than year” are 14, 15 and 16 respectively. Likewise, $R(tg_Q')$ is the rank of tg_Q'.

7.3 Experimental Results and Analysis

In the experiments, language model is taken as the baseline model. Performance of SNoW-based classifier will be compared with that of language model. Different combinations of features are tested in SNoW-based classifier and their performances are investigated.

7.3.1 LM Classifier

The LM classifier takes two types of features: unigram and bigram. Experiment results are presented in Fig. 2.

Accuracy varies with different feature weight λ and best accuracy ($accuracy_1$ 68.0% and $accuracy_2$ 68.9%) achieves when λ=0.7. Accuracy when λ=1.0 is higher than that when λ=0. It indicates that, in the framework of language model, unigrams achieves better performance than bigrams, which accounts from the sparseness of bigram features.

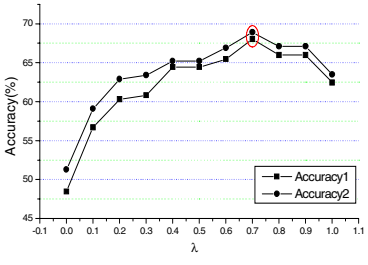


Fig. 2. Accuracy of LM classifier. Data in circle is the best performance achieved.

7.3.2 SNOW Classifier

Our SNoW classifier requires binary features. We then encode each feature with an integer label. When a feature is observed in a question, its label will appear in the extracted feature set of this question. There are six types of features: 15 user preferences (10 for the estimation of time granularities, 3 for the estimation distributions, and 2 for question focuses) (F_1), 951 unigrams (F_2), 9277 bigrams (F_3), 10 granularity of time expressions (F_4), 14 special patterns (F_5), and 4 event types (F_6). Although the number of all features is more than ten thousand, the features in one question are no more than twenty in general. Accuracies of SNoW classifier on 194 test questions are presented in Table 1. It shows that simply using unigram features, SNoW classifier has already achieved better accuracy than LM classifier ($accuracy_1$: 69.5% vs. 68.0%; $accuracy_2$: 70.3% vs. 68.9%). From this view, SNoW classifier outperforms LM classifier in handling sparse features. When all the six types of features are used, SNoW classifier achieves 83.5% in $accuracy_1$ and 83.9% in $accuracy_2$, almost close to the accuracy of user tagging, i.e. 86.2%.

Table 1. Accuracy (%) of SNoW classifier

Feature Set	F_2	$F_{2,3}$	$F_{1\sim6}$
$Accuracy_1$	69.5	72.1	83.5
$Accuracy_2$	70.3	72.7	83.9

Table 2. $Accuracy_1$ (%) on different types of time granularities

TG	second	minute	hour	day	weekday
$Accuracy_1$	100	100	100	64.2	100
TG	week	month	season	year	century
$Accuracy_1$	100	60	100	90.5	66.7

Table 3. Accuracy (%) on combination of different types of features

Feature Set	$F_{2,3}$	$F_{1,2,3}$	$F_{2,3,4}$	$F_{2,3,5}$	$F_{2,3,6}$
$Accuracy_1$	72.1	79.8	73.7	74.7	72.6
$Accuracy_2$	72.7	80.6	74.7	75.2	73.1

With all the six types of features, $accuracy_1$ on the questions with different types of time granularity is illustrated in Table 2. It reveals that the classification errors mainly come from time granularity of "month", "day" and "century". Low accuracy on "month" and "century" accounts from absence of enough examples, i.e. examples for training and testing both less than five. Many "day" questions are incorrectly classified into "year", which accounts for the low accuracy on "day". The reason lies in that there are more "year" questions than "day" questions in the training question corpus (116 vs. 56).

In general, we can extract three F_1 features, one F_4 feature, less than two F_5 features, and one F_6 feature from one question. It is hard for SNoW classifier to train and test independently on each of these types of the features because of the small feature number in one example question. However, the numbers of F_2 and F_3 features in a question are normally more than ten. So we take unigrams (F_2) and bigrams (F_3) as the basic feature set. Table 3 presents the accuracy when the rest four types of features are added into the basic feature set respectively. As expected user preference makes the most significant improvement, 7.82% in $accuracy_1$ and 7.90% in $accuracy_2$. Special patterns also play an important role, which makes 2.6% $accuracy_1$ improvement. It is strange that event type makes such a modest improvement (0.5%). After analyzing the experimental results, we find that as there are only four event types, it makes limited contribution to 10-class time granularity classification.

8 Conclusion

Various features for time granularity classification of temporal questions are investigated in this paper. User preference is shown to make a significant contribution to classification performance. SNoW classifier, combining user preference, word

N-grams, granularity of time expressions, special patterns and event types, achieves 83.5% accuracy in classification, close to manually tagging accuracy 86.2%.

Acknowledgement

This project is partially supported by Hong Kong RGC CERG (Grant No: PolyU5181/03E), and partially by CUHK Direct Grant (No: 2050330).

References

1) TREC (ed.): The TREC-8 Question Answering Track Evaluation. Text Retrieval Conference TREC-8, Gaithersburg, MD (1999)
2) Radev D. and Sundheim B.: Using TimeML in Question Answering. http://www.cs.brandeis.edu/~jamesp/arda/time/documentation/TimeML-use-in-qa-v1.0.pdf, (2002)
3) Li, X. and Roth, D.: Learning Question Classifiers. Proceedings of the 19th International Conference on Computational Linguistics (2002) 556-562
4) S. Abney, M. Collins, and A. Singhal: Answer Extraction. Proceedings of the 6th ANLP Conference (2000) 296-301
5) Saquete E., Martínez-Barco P., Muñoz R.: Splitting Complex Temporal Questions for Question Answering Systems. Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (2004) 567-574
6) Diaz, F. and Jones, R.: Temporal Profiles of Queries. Yahoo! Research Labs Technical Report YRL-2004-022 (2004)
7) Wei Li: Question Classification Using Language Modeling. CIIR Technical Report (2002)
8) Dell Zhang and Wee Sun Lee: Question Classification Using Support Vector Machines. Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (2003) 26-32
9) Jun Suzuki, Hirotoshi Taira, Yutaka Sasaki, and Eisaku Maeda: Question Classification Using HDAG Kernel. Proceedings of Workshop on Multilingual Summarization and Question Answering (2003) 61-68
10) Li X., Roth D., and Small K.: The Role of Semantic Information in Learning Question Classifiers. Proceedings of the International Joint Conference on Natural Language Processing (2004)
11) Schilder, Frank & Habel, Christopher: Temporal Information Extraction for Temporal Question Answering. In New Directions in Question Answering. Papers from the 2003 AAAI Spring Symposium TR SS-03-07 (2003) 34-44
12) Rohini K. Srihari, Wei Li: A Question Answering System Supported by Information Extraction. Proceedings of Association for Computational Linguistics (2000) 166-172
13) Eduard Hovy, Laurie Geber, Ulf Hermjakob, Chin-Yew Lin, and Deepak Ravichandran: Towards Semantics-Based Answer Pinpointing. Proceedings of the DARPA Human Language Technology Conference (2001)
14) Hermjacob U.: Parsing and Question Classification for Question Answering. Proceedings of the Association for Computational Linguists Workshop on Open-Domain Question Answering (2001) 17-22
15) Ittycheriah, Franz M., Zhu W., Ratnaparki A. and Mammone R.: Question Answering Using Maximum Entropy Components. Proceedings of the North American chapter of the Association for Computational Linguistics (2001) 33-39