Extraction of Chronological Statistics Using Domain Specific Knowledge

Tatsukuni INOUE Takashi YAMAMOTO Makoto TORIYABE
Erina SHIMIZU Hiroya SUSUKI Hiroaki SAITO
Keio University
3-14-1 Hiyoshi, Kohoku-ku, Yokohama-shi, Kanagawa 223-8522, Japan
Email: {inoue,yamataka,tori,shimizu,susuki,hxs}@nak.ics.keio.ac.jp

Abstract

This paper reports a system constructed for our participation as a group of "keio01" of Keio University in the T2N (text to number) task at the NTCIR-7 MuST (Multimodal Summarization for Trend Information) task. The constructed system uses newspaper article corpora, task description and domain specific knowledge, and the system outputs chronological statistics. The statistics are ternary data which are pairs of a statistic name and pairs of date and value. They are visualized as drawing charts. The system was evaluated by comparison with manually extracted data, and it achieved 0.785 F-score.
Keywords: *Trend Information, Statistics Extraction.*

1 Overview of Our System

Our system uses as input an XML file of the T2N task description, original raw newspaper articles and domain specific knowledge to be described. The system extracts pairs of date and value which correspond to the query, that is also the name of chronological statistics, specified in the task description. Hereafter, this paper describes the pair of a *query* and pairs of *date* and *value* as *ternary data*.

In the articles, the extractable data are written in two ways. Some data are explicitly written in the articles, and some are obtained by calculation and/or reasoning using explicit expressions. Our system extracts the former only.

Here a specific example of extraction of statistics is shown. If the query is "レギュラーガソリンの全国平均店頭価格" (pump price of regular gasoline (national average)), then a pair of date "2000 年 10 月 16 日" (16th October 2000) and value "105 円" (105 yen) becomes a candidate of extraction, and the data like this makes it possible to visualize the chronological statistics by drawing a chart.

2 The Queries

The queries are stated in overview of MuST at the NTCIR-7 workshop.

3 Domain Specific Knowledge of Statistics

In the task description, although the domain specific knowledge, which consists of both *numerical unit names* and *aliases* (alternative names) for the queries, is given, we used additional knowledge shown below. From now on, this paper labels these domain specific knowledge as *knowledge of statistics* collectively.

Figure 1 shows the relation of inputs (a query and the target articles given in the task description), domain specific knowledge shown below and knowledge of statistics as entire knowledge.

3.1 Indispensable Terms

An *indispensable term* is the word that should be structurally dependent on a term which denotes the value of a ternary data. For example, if the query is "レギュラーガソリンの全国平均店頭価格", then the indispensable term is "ガソリン" (gasoline).

There may be multiple indispensable terms in a paragraph, where all or some of them are dependent on the value term.

We included these indispensable terms into the knowledge of statistics and utilized the knowledge on how many terms should be dependent on the value term.

3.2 Disallowed Terms

A *disallowed term* is the word that should NOT be structurally dependent on a term which denotes the value of statistics by the same dependency structure analysis as one mentioned in Section 3.1. For example, if the query is "レギュラーガソリンの全国平均

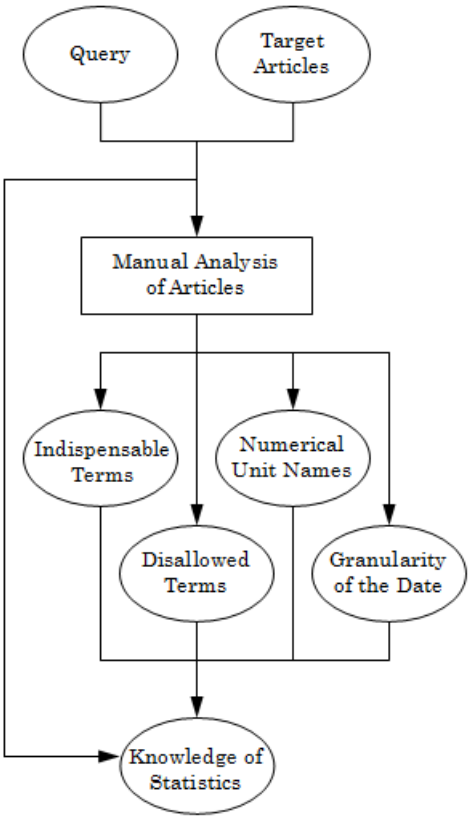

Figure 1. Relation of Inputs and Knowledge of Statistics

店頭価格”, then the disallowed term is “軽油” (light oil).

When the disallowed term is dependent on the value regardless of dependency of the indispensable term, the value is identified as the data which should not be extracted. In addition, if multiple disallowed terms are given for the query and one of them is dependent on the value term, it is also identified as the data which should not be extracted.

We also utilized the disallowed terms as the knowledge of statistics.

For instance, if “ガソリン” is the indispensable term and “軽油” is the disallowed term, then only “１０３円” is accurately extracted from the following sentence.

> 日本国内でも原油価格上昇を反映し、今月２日現在のガソリンの店頭価格（１リットル当たり）が全国平均で１０３円、軽油が８３円と、先月に比べそれぞれ１円値上がりした。

3.3 Aliases of Numerical Unit Names

Depending on the query, there are unextractable values using only numerical unit names given in the task description. For instance, if the query is “ＰＨＳの加入台数” (the number of PHS subscribers), although there is “台” as the unit name given in the task description, there are “件”, “加入”, “人” as unit name expressions in the newspaper articles. Therefore we put the aliases of the unit names in knowledge.

For some query, there is no unit name in the value term in rare cases. Thus, we utilized the unit names in the query-unrelated value terms as knowledge.

3.4 Granularity of the Date

The granularity of extractable dates for each query varies, for instance, “day”, “month”, “quarter year”, “year”. Depending on the query, the granularity is not unique and, furthermore, there are inappropriate granularities of the dates. For example, if the query is “センター試験の志願者数” (the number of applicants of the Center Exam), then the granularity of dates should not be “day” or “month” but “year”. Therefore we utilized the granularity of dates.

4 Used Dictionaries

The system uses dictionaries for arbitrary queries separately from the knowledge described in Section 3. There are two kinds of dictionaries: for comparison expressions and for date expressions.

4.1 Comparison Expression Dictionary

The comparison expression dictionary is used for extracting comparison values. Comparison expressions compares one data to another. An example comparison expression is “前週に比べ１円上昇した” (be up by 1 yen from last week). In this example, “１円” is the obviously value, but the value denotes a difference between values. This paper calls the value as a *comparison value*.

It is possible to calculate statistics indirectly using the comparison values. However, in our system, comparison expressions are used in later processing stage. Details are described in Section 5.4.

In most cases, the comparison values are structurally dependent on the comparison expressions which denote changes in statistics. Therefore the system uses the manually written dictionary of the comparison expressions which were extracted from the specified articles in the task description.

4.2 Date Expression Dictionary

The date expression dictionary is used for both extracting date expressions and estimating the absolute date the expressions mean. Date expressions denote time points. The date expression dictionary is also

created manually in a similar way of the comparison expressions. The date expressions are extracted from the specified articles in the task description.

If a date expression has digits, for example "１９９７年６月" (June 1997), the digit part of the expression is registered as variables on the dictionary. If the expression means the relative date to the date of the article, for example "先月" (last month), the rules of estimating the absolute dates are also registered on the dictionary.

5 System Configuration

This section shows the flow of our system of extracting ternary data. The constructed system utilizes the knowledge of statistics previously described, and the system uses CaboCha[†] for dependency structure analysis of Japanese texts.

Figure 2 shows the flow of the system. The extraction process consists of two stages. The first stage is estimating the range of values using the entire newspaper article corpora, and the second stage is extracting ternary data from articles specified in the task description. In the second stage, the system uses the range estimated in the first stage, the domain specific knowledge mentioned in Section 3 and the dictionaries described in Section 4.

5.1 Identifying the Query Related Articles

In this task, although the target articles of extracting ternary data are specified in the task description, the system identifies the articles relating to the query from Mainichi Newspapers' articles from 1998 to 2001.

These articles are identified under the condition that the articles contain the indispensable terms satisfying the conditions described in Section 3.1. This identification is performed because the processing time can be reduced for the procedure described in Section 5.2.

5.2 Estimation of the Range of Values

Our system extracts the values of candidate ternary data using both the articles identified in Section 5.1 and the knowledge which are indispensable terms, disallowed terms and numerical unit names mentioned in Section 3. Moreover, using the extracted values, the system estimates the range of values of ternary data. The following shows the method of estimating the range of values.

First, the system extracts the values of candidate ternary data as follows:

1. Extract sentences including conjunction of "value" and "numerical unit name" (for example, "１０５円") from the articles described in Section 5.1.
2. Extract conjunctions of "value" and "unit name" from the extracted sentences, then make pairs of the conjunction and the extracted sentences which were dependency-structure-analysed.
3. Eliminate the pairs whose conjunction is structurally dependent on a comparison expression (described in Section 5.4) in the sentence.
4. Extract the "value" as the value of ternary data from the pairs whose conjunction is structurally dependent on the indispensable terms in the sentence.

Using the procedure above, for instance, toward the sentence quoted in Section 3.2, the system extracts "１０３円" and "８３円" from the sentence. In this procedure, the system does not use the knowledge of disallowed terms because the number of the articles identified in Section 5.1 becomes significantly large and therefore it is difficult to manually extract disallowed terms from all the articles.

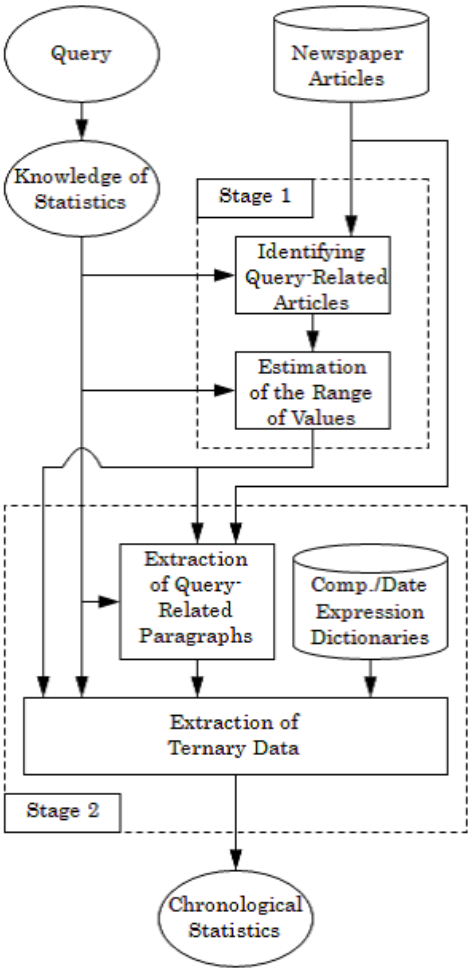

Figure 2. System Configuration

[†]http://chasen.org/~taku/software/cabocha/

Secondly, the system estimates the range of values as follows:

1. Sort the extracted values in an ascending order.
2. Calculate common logarithm of the values, then calculate difference of adjacent values.
3. Detect positions of the differences which exceed a predetermined threshold value, then identify the positions as *boundaries*.
4. Regard positions both before the first value and after the last value as imaginary boundaries.
5. Calculate the number of the values in the regions between the boundaries, then detect a region where the number is maximum.
6. Extract the minimum/maximum values in the detected region, then calculate the product of the values and the correction factors.
7. Set the calculated products as the lower and upper limits of the range of values.

The reason the system calculates common logarithm of the values in Step 2 is that if the ratio between adjacent two values exceeds a certain threshold and either of the two values is the correct value of statistics then there is a high possibility that the other value should not be extracted.

Meanwhile, threshold t used in Step 3 and the correction factors of minimum/maximum values $r_{\min}$, $r_{\max}$ used in Step 5 are calculated empirically from the following formulae. The formulae are shared by all the tasks. In the formulae, N is the number of values of candidate ternary data.

$$t = 0.15{+}0.3/\log_{10}(N),\ \ r_{\min} = 1/10^{t},\ \ r_{\max} = 10^{t}$$

Here a hypothetical example is given. If the extracted values are all five of "100", "1000", "2000", "5000" and "20000", then t becomes 0.579, the minimum/maximum values between the detected boundaries become "1000" and "5000", finally, the lower and upper limits of the range become "263" and "18975".

If the numerical unit name is "%", the range is obviously from 0 to 100. Therefore, in this case, the system does not estimate the range using the method above.

5.3 Extraction of the Query Related Paragraphs

Using the aforementioned knowledge of statistics and the automatically acquired information, the system extracts the paragraphs that might include ternary data from the newspaper articles specified in the task description. The following shows the extraction procedure.

1. Extract the paragraphs including the conjunction of "value" and "numerical unit name" from the specified articles.
2. Eliminate the paragraphs where all the values of the conjunctions are beyond the range estimated in Section 5.2.
3. Extract the paragraphs including the indispensable terms.

For instance, the following paragraph is extracted in our system.

> また、携帯電話より通話料金で割安感のあるＰＨＳは同月、１万９０００台が新規に加入。加入者総数は５８４万２０００台となり、３年ぶりに前年同期の実績（５７０万７０００台）を上回った。携帯電話とＰＨＳを合わせた加入者数は６６７８万３０００台で、前年同期（５６８４万６０００台）と比べ１７・５％増、普及率も５２・６％に達した。【野島康祐】

5.4 Extraction of Comparison Expressions

The system extracts comparison expressions from the paragraphs extracted in Section 5.3, and produces the paragraphs where the values in the comparison expressions were automatically-tagged. The system extracts the comparison values on the condition that the extracted paragraphs include the expression in the comparison dictionary described in Section 4.1 and the expression is dependent on the value.

The following paragraph shows an example of extracting comparison values. The system uses tag notations which are uniquely redefined in the MuST corpus. In the notations, when `type` is `"diff"/"prop"`, the comparison value means difference/ratio between the values. When `updown` is `"u"/"d"`, the comparison value means rise/fall from the previous value.

> また、携帯電話より通話料金で割安感のあるＰＨＳは同月、<rel type="diff" updown="u">１万９０００台が</rel>新規に加入。加入者総数は５８４万２０００台となり、３年ぶりに前年同期の実績（５７０万７０００台）を上回った。携帯電話とＰＨＳを合わせた加入者数は６６７８万３０００台で、前年同期（５６８４万６０００台）と比べ<rel type="prop" updown="u">１７・５％増、</rel>普及率も５２・６％に達した。【野島康祐】

5.5 Extraction of Date Expressions

The extraction process of the date expressions has two phases. One is extracting the expressions which

denote time points, and the other is estimating the date the extracted expressions mean.

In the former phase, the system uses the date expression dictionary described in Section 4.2. The system searches the specified articles for the date expressions in the dictionary, and extracts the expression. In the latter phase, the system estimates the absolute dates using the pair of the extracted date expression and the date of the article including the expression.

The system extracts the date expressions from the paragraphs auto-tagged to the comparison values in Section 5.4 and then automatically tags the extracted date expressions.

The following paragraph is an example of the extraction of the date expressions. The `date` tag means the point of time and the `dur` tag means the duration time. The attribute `gra` shows the granularity of the date and the attribute `abs` shows the absolute date (year, month, day) the system estimated. If a date expression has the `dur` tag, the date estimated from the expression does not become the date of ternary data.

```
　また、携帯電話より通話料金で割安感のあるＰＨＳは同月、<rel type="diff" updown="u">１万９０００台が</rel>新規に加入。加入者総数は５８４万２０００台となり、<dur gra="月" abs="199804">３年ぶりに</date><date gra="月" abs="200004">前年同期の</date>実績（５７０万７０００台）を上回った。携帯電話とＰＨＳを合わせた加入者数は６６７８万３０００台で、<date gra="月" abs="200004">前年同期</date>（５６８４万６０００台）と比べ<rel type="prop" updown="u">１　７・５％増、</rel>普及率も５２・６％に達した。【野島康祐】
```

5.6 Extraction of the Pairs of Date and Value

Using both the acquired information and domain specific knowledge described in Section 3, the system can extract ternary data, which is the goal of the task. This section shows the procedures to extract ternary data.

5.6.1 Extraction of the Values

First, the system extracts the values from the paragraphs auto-tagged to the date expressions in Section 5.5 as follows.

1. Extract the conjunctions of "value" and "numerical unit name" from the auto-tagged paragraphs.
2. Eliminate the conjunctions which were tagged as comparison expressions or date expressions.
3. Extract the conjunctions which are structurally dependent from the indispensable terms (described in Section 3.1) in the paragraph including the conjunction.
4. Eliminate the conjunctions which are structurally dependent from the disallowed terms (described in Section 3.2) in the paragraph including the conjunction.
5. Extract the values in the range estimated in Section 5.2 from the extracted conjunctions.

5.6.2 Extraction of the Date Expressions Combined with the Values

Second, the system extracts the date expressions combined with the values extracted in Section 5.6.1. The date expressions are searched from both the parse tree obtained by dependency structure analysis of the paragraph and the entire article including the value by the following two algorithms. The algorithms are applied in order of A and B.

A Extraction from the parse tree of the paragraph

1. Search the parse tree node which is a phrase including the value (hereinafter called "start phrase"), then set the node to the target phrase of search.
2. If the target phrase includes the date expressions, then complete the search by extracting the target.
3. If there are phrases being structurally dependent on the target phrase and not having been searched, then set the phrase, in an ascending order of the distance from the target, to the next target and then go back to Step 2.
4. Set the phrase on which the target node is structurally dependent to the next target.
5. If the start phrase is the "ハ" case and the target phrase appears in the phrases before the end of the sentence including the target or the start phrase is not the "ハ" case and the target phrase appears before the start phrase, then go back to Step 2.
6. If there are phrases not having been searched before the start phrase, then set the phrase being nearest from the start phrase to the next target and go back to Step 2. Otherwise, give up searching the date expression.

B Extraction from the entire article

1. Set the phrase at the beginning of the body text of the article to the target of search.
2. If the target phrase has been searched using Algorithm A, then give up the search.

3. If the target phrase includes the date expression, then complete the search by extracting the target.

4. Set the phrase adjacent to the target to the next target, and go back to Step 2.

Depending on the query, there are a lot of values which should not be extracted in the article. In this case, if the system searches the date expressions from the entire article, the system might extract incorrect dates. Therefore, depending on the query, the system does not apply Algorithm B. Whether the system applies Algorithm B is given as domain specific knowledge.

In addition, when the system extracts date expressions, the system utilizes various rules. For example, in extracting dates from the paragraph obtained in Section 5.5, although there are date expressions including the expression of "同期" (equivalent period) in the parse tree nodes of the sentence, the system does not extract the date expressions as a general rule. As a result, "６６７８万３０００台", that denotes "携帯電話とＰＨＳを合わせた加入者数" (the total number of PHS and cellular phone subscribers), is regarded as the value in April 2001, and it is extracted applying the Algorithm B. (The exact date is March 2001. See the article of ID: 010407021.)

However, using the general rule, the system cannot extract the exact dates of "５７０万７０００台" as the value of "ＰＨＳの加入者", and "５６８４万６０００台" as the value of "携帯電話とＰＨＳを合わせた加入者数". Therefore, if a date expression including "同期" lies adjacent to a phrase of the value, which includes a numerical unit name corresponding the query, then the date is extracted as an exception.

6 Evaluation

The result of extracting ternary data from Mainichi Newspapers' articles specified in the task description using the system described in this paper was evaluated by comparison with manually extracted data.

There were 314 correct ternary data in the specified articles. The system extracted 239 ternary data out of which 217 data were collect. As a result, the system achieved 0.908 precision, 0.691 recall and 0.785 F-score.

7 Discussion

This section discusses the reasons why the system extracted incorrect data and the system could not extract correct data. The former reasons mean factors reducing precision, and the latter reasons mean factors reducing recall.

7.1 Factors Reducing Precision

The extraction of incorrect data by the system is largely due to the following two reasons.

As the first reason, there are statistics which are either superset or subset of the specified statistics in the specified articles. For instance, if the query is "パソコン国内出荷台数" (the volume of national shipments of personal computers), there are "パソコンの国内外への総出荷台数" (the total volume of national and international shipments of personal computers) as a superset of the statistic and "ノートパソコンの国内出荷台数" (the volume of national shipments of notebook computers), as a subset of the statistic. Because it is difficult to differentiate them from the statistic of the query, the system might extract incorrect data.

The second reason is that there are date expressions being difficult to estimate the absolute date. One of the most difficult examples is the date expression "上半期" or "下半期". Both of these possibly mean two periods of time, and if the expression is "上半期", it might mean either "first half of the *calender* year" or "first half of the *fiscal* year". In the former case, the period is from January to June. In the latter case, the period is from April to September. The system does not discriminate them, and regards the expression as the latter.

7.2 Factors Reducing Recall

The failure of extracting correct data by the system is due to the following reasons.

First, some values, which should be extracted, are not structurally dependent from the specified indispensable terms. In this case, the system can not extract the values. This problem occurs when descriptions of the values are abbreviated.

The following paragraph includes unextractable values: "３６０万５０００" and "４８０万台" as "ｉモード加入者数" (the number of i-mode subscribers). Since the paragraph does not include the indispensable term ("ｉモード"), the system cannot extract the paragraphs as described in Section 5.3.

> 昨年２月下旬にスタートし、１年足らずで加入台数は３６０万５０００を突破。３月末目標は何度か上方修正を繰り返し、４８０万台に。ドコモが用意した３１２の公認ホームページのほか、ボランティアサイトと呼ばれる自主ホームページも約４８００に上る。競合他社も同様のサービスを手掛け、携帯電話市場の主戦場になっている。

Second, the number of domain specific knowledge for ternary data extraction is not enough. Because the knowledge to extract the data is appended in order of decreasing effectiveness for the extraction the knowledge to extract difficult data is not enough. Therefore

the knowledge has an aspect of being effective for improving precision, but not for improving recall.

The knowledge of “内閣支持率” (approval rating for the Cabinet) is a good example of the problem above. The system needs various knowledge to extract the statistics with high precision and recall. It is particularly difficult to collect the knowledge for high recall extraction. Consequently, the system extracts the data from paragraphs including the term “全国世論調査” or “全国電話世論調査” (national opinion poll) for high precision. As a result, the system obtained 0.481 recall and 0.929 precision.

Third, the system does not extract the data from the headline of articles. Since the headline has properties differing from the body text and the data which the headline includes is extractable from the body text, the mechanism of the extraction from headlines was not added to the system. The T2N task treats data in headlines as separate data from body texts, even if data in headline means the same data as in the body text. Thus if a system does not extract the data in headlines, recall is reduced.

For instance, while the following headline includes ternary data, the system extracts the same data from the body text of the article of the headline. There are 7 correct data, including the instance, being unextractable because of the reason above.

移動電話、「一般加入」上回る－－３月末で５６８５万台、普及率４４．８％に

Fourth, there are harmful effects of estimating the range described in Section 5.2. Though using the range of values improves precisions of the extracting some statistics, the data of values beyond the range are eliminated from the candidate data. Therefore, depending on the result of estimating the range, the range makes recall of extracting statistics worse.

For instance, the range of the values of “ｉモード加入者数”, which is estimated using our proposed method, is from 436471 to 158773322. As a result, when value falls below the lower limit of the range (for example, “２５万台”), the value becomes unextractable.

7.3 Effect of Estimating the Range

As discussed in the end of Section 7.2, estimating the range of values yields two contrary effects: precision improvement and recall reduction. The following describes the change in precision and recall caused by estimating the range.

In analyzing queries, the numerical unit name of the statistics excludes “%”. The reason is that if the unit name is “%”, the range is obviously from 0 to 100. We analyzed the change caused by whether the system uses the range in the processes described in Section 5.3 and 5.6.1.

Table 1 shows the analysis result. If the system uses the range of values, then, although the recall is slightly reduced, the precision is greatly improved. The F-score also rises by approximately 20 percent.

Table 1. Evaluation of the system using the range

Range	Precision	Recall	F-score
Not Used	0.632	0.748	0.685
Used	0.914	0.743	0.820

The reason the recall is reduced is that the system cannot extract the only one data (“ｉモード加入者数” is “２５万台”) described in Section 7.2. The reason the degree of the reduction is small is due to the configuration that the formula to calculate both the threshold value and the correction factors mentioned in Section 5.2 gives quite wide ranges.

On the other hand, the reason the degree of precision improvement is large is due to the statistics which have no numerical unit name. Depending on statistics, there is no unit name (e.g. “鉱工業生産指数” (industrial production index)), or the unit name is abbreviated in the articles (e.g. “ＰＨＳの加入台数”). In the case of extracting the statistics above, if the system does not use the range, the system needs to extract the morphemes following digits and to differentiate the unit name of the statistics from the morphemes. However the system does not perform the procedures above because the incorrect data extracted in this case is eliminated using the method described in Section 5.6.1. Therefore, for example, if the query is “鉱工業生産指数”, the system also extracts the values including the unit name “%” regardless of whether the system uses the range of values.

As a result, in the case of not using the range, if the system performs the differentiation above, precision may be improved. However there are also incorrect data that the system can eliminate only using the range. For example, if the query is “デジカメの国内出荷額” (the value of national shipments of digital cameras), the system eliminates “１０万円” which means the price of digital cameras using the estimated range.

8 Related Works

Nanba et al. [1] used 26 kinds of comparison expressions which are manually selected to extract comparison values. In our system, comparison expressions, that are also manually selected, similar to the above are used.

The system constructed by Nanba et al. made an assumption that when the system extracts time expres-

sions, the granularity of the dates to be extracted are previously given by the users. In our system, using the knowledge of the granularity of the dates described in Section 3.4 is similar to the assumption above.

In addition, the system constructed by Nanba et al. complements the abbreviation of time expressions. This is a processing that if the date expression (for example, month, year) is abbreviated, the system estimates the absolute date using both the date expression and the date of the article including the expression. Our system also performs the same processing as above.

In Soga's system [2], when the system extracts the pair of date and value, the system extracts the date expressions from the sentences including data by using the priority based on the relationship of dependency structure between the date expression and the value expression. Our method to extract the pair is newly-designed by reference to the processing above.

9 Future Work

A direction to develop the current system is to reconstruct a more versatile system which does not depend on the domain specific knowledge described in Section 3. The current system needs manually given knowledge corresponding to the query specified in task descriptions. The time required to pick out the knowledge is accordingly large. Therefore the current system is hard to be called an automatic system. There are two directions to solve this problem.

The first is automatic acquisition of the knowledge. In this case, the acquired knowledge needs to be concretized and the system needs to adopt the method to extract the knowledge accurately.

The second is designing a new method without either acquiring the knowledge or heavily depending on the knowledge. A specific example is the improvement of estimating the range of values described in Section 5.2. If the system can estimate the range with high accuracy, it will be possible to reduce needed knowledge.

References

[1] Hidetsugu Nanba, Yoshinobu Kunimasa, Shiho Fukushima, Teruaki Aizawa, Manabu Okumura: "Extraction and Visualization of Trend Information Based on the Cross-document Structure" (in Japanese), IEICE technical report, Vol.105, pp.67-74, 2005.

[2] Masaya Soga, Hiroaki Saito: "動向情報提示システムの構築" (in Japanese), Proceedings of the Workshop "言語処理と情報可視化の接点" on the 12th Annual Meeting of the Association for Natural Language Processing, pp.5-8, 2006

Appendix

"Keio02" System

This appendix briefly reports on our "keio02" system in the NTCIR MuST T2N task. The Mainichi corpus was preprocessed and annotated in advance; phrases denoting time are identified by manually written 48 regular expression rules. Numeral figures are identified as values along with its unit name.

The annotated sentences are then classified into the most related query name by manually written 130 rules which utilize the neighbouring words and near phrases. In that process keio02 assumes that query name locates before its value in the sentence. Keio02 also assumes that the date information of a query name locates before its value, otherwise the date of the article is used as its date. Keio02 puts much emphasis on word cooccurence information overall.

In the formal run keio02 attained 48.0% precision, 41.4% recall and 44.5 F-score. Evaluation showed that keio02 performs well against such query names as mobile phones or i-mode, but was weak in handling cabinet/political party approval rate because various approval/disapproval rates tend to appear in a sentence at the same time.

Critical analysis of the evaluation revealed that a previous sentence should also be taken care of and the (at least syntactic) structure of a sentence should be analyzed.