Synthesis of Multiple Answer Evaluation Measures using a Machine Learning Technique for a QA System

Yasuharu MATSUDA Takashi YUKAWA
Nagaoka University of Technology
1603-1, Kamitomioka-cho, Nagaoka-shi, Niigata 940-2188, Japan
yasuharu@stn.nagaokaut.ac.jp yukawa@vos.nagaokaut.ac.jp

Abstract

The present paper proposes a new method that synthesizes answer evaluation rules using layered neural networks. A Base Question Answering System that employs a combined conventional method (NUT-BASE system) is implemented and evaluated in the NTCIR-5 workshop Question Answering Challenge 3 (QAC3). Based on the evaluation results, the authors focus on performance improvement for the list task and propose a new method using a neural-network-based machine learning technique for synthesizing answer candidate evaluation measures. There are several measures by which to evaluate the likelihood of the answer candidate, so the system must synthesize these measures in order to determine the answer set. However, the rule for synthesizing the measures in the NUT-BASE system was not effective because it was based on an empirical intuition. Therefore, a performance improvement is expected by the proposed method because it is based on quantitative reason. The experimental evaluation showed that the proposed method achieves a performance improvement, with a value of 0.01 for the mean F-measure.

Keywords: *Question Answering System, List Task, Machine Learning, Layered Neural Network*

1. Introduction

As a participant of the NTCIR-5 workshop Question Answering Challenge (QAC) track, the authors, i.e. the NUT (Nagaoka University of Technology) team, implemented a first-stage question answering system (NUT-BASE). The system applies a vector-space model and a phrase attribute analysis technique (Question Focus; QF) [2], and is implemented with newly developed QF-based heuristic rules and information retrieval modules using GETA [1]. The evaluation results show that the NUT-BASE system recorded a value of 0.101 for the mean of the modified F-measure (MF1) [3].

As described above, the NUT-BASE system was comprised of conventional methodologies and newly developed heuristic rules. These rules are based on empirical knowledge of Japanese grammar. From the results, several issues are extracted. Among these issues, poor accuracy of an answer candidate evaluation reduces the system performance for the list task significantly. Therefore, the focus of the present paper is the improvement of the answer candidate evaluation.

In the answer candidate evaluation phase of the QA system, there are several measures of likelihood of answer candidates (ACs). The measures include the position of the ACs in the retrieved documents, the relevance of the QF and the ACs, the number of documents relevant to the ACs, and a number of more detailed measures. The importance of these measures varies depending on the interrogative type of the query and presence of the QF.

In the list task, if the difference between the score for a correct answer set and that for an incorrect answer set is remarkable, then the discriminability of correct answers will rise.

Taking this into consideration, a new evaluation measure synthesis method using the machine learning technique with layered neural networks is proposed and implemented. The present paper describes the details of the proposed method and shows the performance improvement compared with a slightly tuned NUT-BASE (NUT-BASE2) system.

2. Development of the Base QA System

First, as a basis for discussion, the base QA system (NUT-BASE) was developed. This system is comprised of the 'Question Focus' method [2] and newly developed heuristic rules.

Figure 1 shows an overview of the NUT-BASE system. Generally, in a QA system, the answer set is extracted through four phases as follows:

1. Query Analysis Phase,
2. Document Retrieval Phase,
3. Answer Candidate Extraction Phase,
4. Answer Candidate Evaluation Phase.

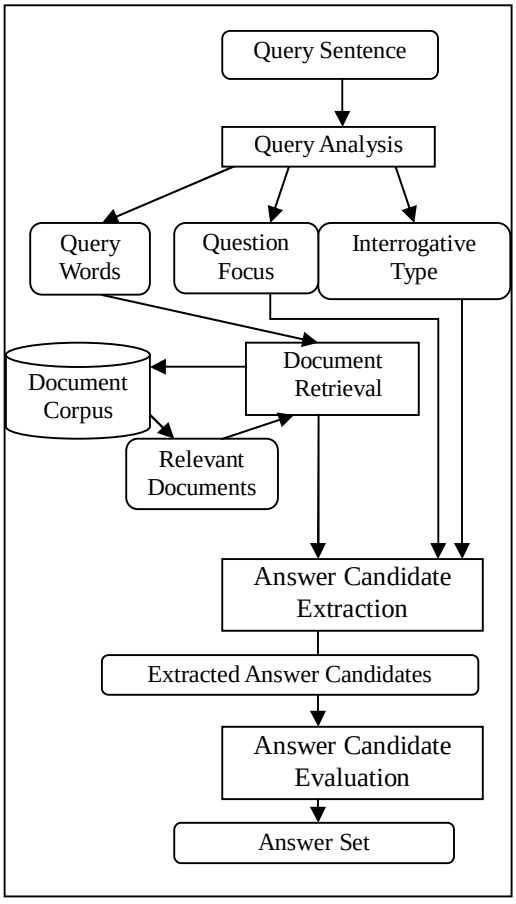

Figure 1. Overview of the base QA system

The NUT-BASE system also follows this general architecture, and so has four modules corresponding to each phase. These modules are described in detail in the following subsections.

2.1 Query Analysis Module

The query analysis module analyzes interrogative types and extracts a QF phrase.

Interrogatives are classified into seven types (what, who, when, where, how, how_many, what-QF), which can be easily distinguished by the interrogative appearing in the query sentence. The type "what-QF" corresponds to the form of "What QF is it?".

QF is the phrase that corresponds to the concept containing ACs. However, excessively abstract phrases such as "thing" and "one" are excluded.

The interrogative type and the QF are extracted by pattern matching with regular expressions.

2.2 Document Retrieval Module

The document retrieval module retrieves documents containing ACs exploiting the vector-space model with the TF-IDF algorithm. The modules are implemented using the GETA [1] library.

2.3 Answer Candidate Extraction Module

The processes of the answer candidate extraction module are comprised of three sub-phases. First, words in the retrieved documents are analyzed as morphemes by the ChaSen parser [4]. Second, some of the words are combined into phrases by NExT [5] and a number of heuristic rules. Finally, the phrases that have attributes corresponding to the interrogative type of the query sentence are extracted ('who' and person's name, 'when' and date or time, etc) as ACs.

2.4 Answer Candidate Evaluation Module

The answer candidate evaluation module gives partial scores depending on the evaluation rules as follows:

- Distance between the AC and the index term of the query in the retrieved document.
- Attribute of the AC.
- Whether a QF is included as a suffix of the AC.
- Whether there is a sentence that includes both the AC and the QF and the AC is an instance of the QF, among the corpus.
- Number of retrieved documents that contain the AC.

These partial scores are synthesized with the newly developed heuristic rules into the final score, and the AC set that has higher score is extracted as the result.

2.5 Results of QAC3

Table 1 shows the results of the QAC3 formal run and the reference-1 run for NUT-BASE system.

Table 1. Results of the QAC3 formal run and the reference-1 run for the NUT-BASE system (MF1)

Query Set	Total	First	Rest
Formal Run	0.101	0.129	0.096
Reference-1 Run	0.116	0.129	0.114

The values are the mean of the modified F-measure (MF1) [3]. The values in the 'Total' column indicate the results of all questions, and those in the 'First' column indicate the results of the first questions of each query series. The values in the 'Rest' column indicate the results of the questions of each query series, excluding the first questions.

These are the official records of the NUT-BASE system in QAC3. As the overall results of the formal run, this system ranked tenth among the 16 systems examined.

3. Evaluation Measure Synthesis with Layered Neural Networks

The NUT-BASE system used in QAC3 had a number of problems. Some of which were caused by the heuristic rules. Thus, a number of heuristic rules were modified to improve the performance. This modified base QA system is referred to as NUT-BASE2.

In addition, a new method was proposed to improve the performance for the list task.

3.1 Concept of the Proposed Method

In the NUT-BASE system, the rule for synthesizing the partial scores in the answer candidate evaluation module was not derived quantitatively, but rather by empirical intuition.

In the phases of distinguishing the interrogative types and extracting the QFs, empirically-derived rules are effective, because these phases are grounded on natural language grammar. On the other hand, humans only have empirical intuition for evaluating multiple partial scores. Therefore, an automatic rule construction method is required for this phase.

If the method can derive a rule for synthesizing the partial scores that gives a higher total score to the correct ACs and a lower total score to the incorrect ACs, then the threshold value that distinguishes the correct and incorrect ACs would be determined more easily and an improvement of system performance for the list task can be expected.

Figure 2 shows an overview of the proposed system.

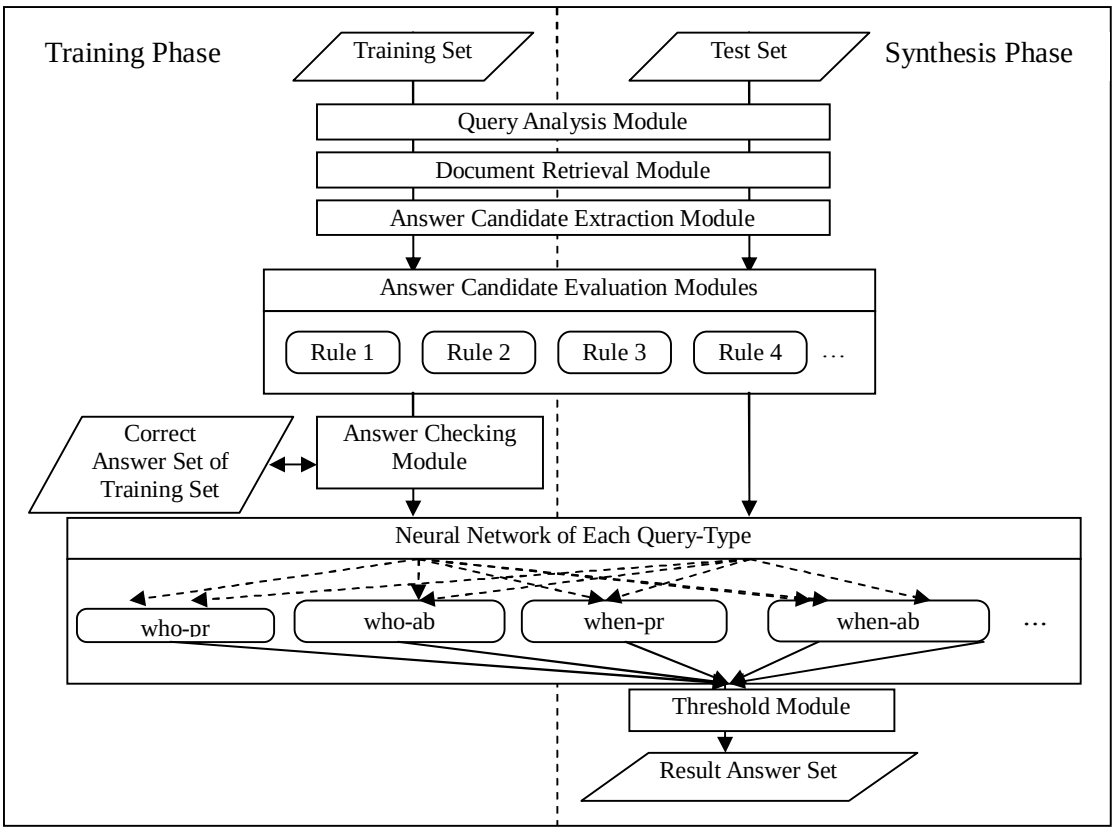

Figure 2. Overview of the proposed system

3.2 Processes of the Proposed Method

The proposed method is comprised of a training phase and a synthesis phase. The training phase includes three processes: distinction of query-type, generation of training data, and training of neural networks. This subsection describes these processes and the process of evaluation measure synthesis in detail.

In the proposed method, the terms related to query sets are defined as follows:

Training Set: A query set used in the training phase to generate the training data. Correct answer sets of each query are known.

Test Set: A query set used for evaluating the system performance.

3.2.1 Distinction of Query-Type

A neural network is defined for each Query-Type. The Query-Type is the combination of interrogative type and presence of the QFs.

The query analysis module distinguishes the interrogative types from an interrogative in the query. Then, these interrogative types are classified further by the presence of a QF. If a QF is present in the query sentence, then the Query-Type is expressed in the form '-pr' after the interrogative type. If a QF is absent, then the Query-Type is expressed in the form '-ab' after interrogative type. However, there are some exceptions to these definitions. As described in Section 2.1, if the form of the query sentence is "What QF is it?", then the interrogative type is defined as 'what-QF'. If the interrogative is 'where' and a QF is present in the query, then the module distinguishes the subtype of the query according to the attribute of the QF. If the QF is a word indicating a place, then the Query-Type is 'where-loc'. If the QF is a word indicating an organization, then the Query-Type is 'where-org'.

Table 2. Query-Type definition

Interrogative Type	Presence of QF	
	present	absent
who	who-pr	who-ab
when	when-pr	when-ab
how	how-pr	how-ab
how_many	how_many-pr	how_many-ab
where	where-loc where-org where-pr	where-ab
what	what-pr what-QF	what-ab

If the QF cannot be distinguished, then the Query-Type is 'where-pr'.

Table 2 shows all 15 patterns of Query-Types. The system contains 15 neural networks. Each neural network corresponds to a Query-Type.

3.2.2 Generating Training Data

The system generates sets of training data for the 15 Query-Types described above.

The training set is input to the system and processed in the query analysis module, the document retrieval module, and the answer candidate extraction module in the same way as those in the NUT-BASE system. Then, the answer evaluation rules in the answer candidate evaluation module give the partial scores to ACs. The values of the partial scores range from 0 to 1.

Then, the answer checking module checks these ACs with the known correct answer set of the query, whether they are correct or incorrect. The module gives a value of 1 to a correct AC and 0 to an incorrect AC.

A tuple containing the partial scores and an answer checking result for each AC forms an element in the training data. Figure 3 shows the processes of generating training data.

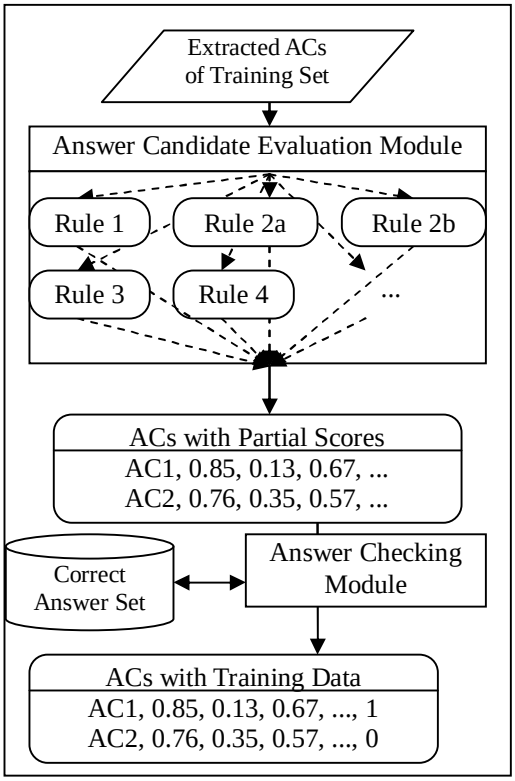

Figure 3. Generating training data

3.2.3 Training the Neural Networks

The system trains the layered neural networks

that correspond to each Query-Type by the generated training data. The layered neural networks use the Back-Propagation method for training [6]. Figure 4 shows the processes of training the layered neural networks.

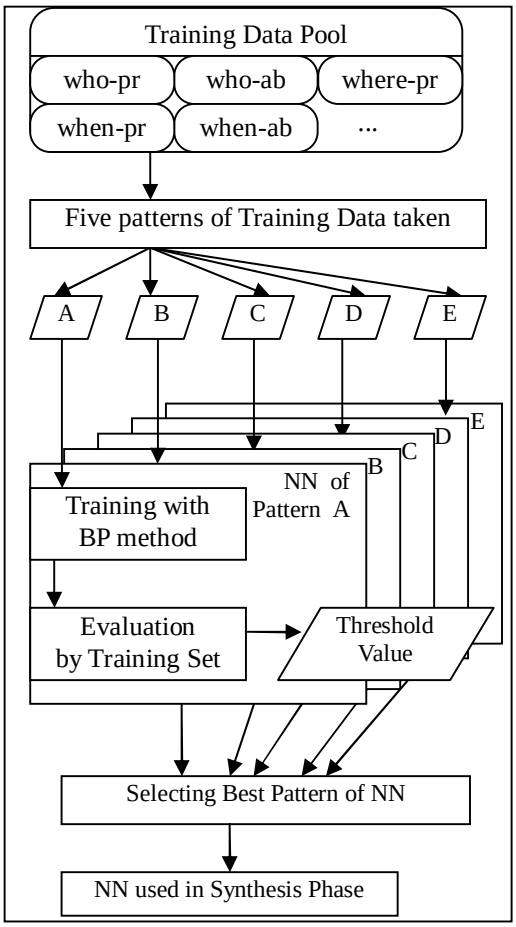

Figure 4. Training the neural networks

For each Query-Type, the neural networks are trained using five patterns of subset of training data that are taken from the training data generated by the processes described above. The patterns of the training data subset are described as follows:

A: All of the elements
B: 1,000 elements taken at random
C: 5,000 elements taken at random
D: 10,000 elements taken at random
E: All elements of correct answers + 500 elements of incorrect answers taken at random

After the neural networks are trained, for each Query-Type, the system evaluates each pattern of trained neural network using the training set as input queries and selects the neural network that marks the highest F-measure and the threshold value for determining the answer set. The training phase is finished when the neural networks and the threshold values of all Query-Types are selected. The system is then ready to answer queries. These neural networks and threshold values are used in the evaluation measure synthesis module at the synthesis phase.

3.2.4 Evaluation Measure Synthesis

The trained neural networks in the training phase are used in the evaluation measure synthesis module. Figure 5 shows the processes of evaluation measure synthesis.

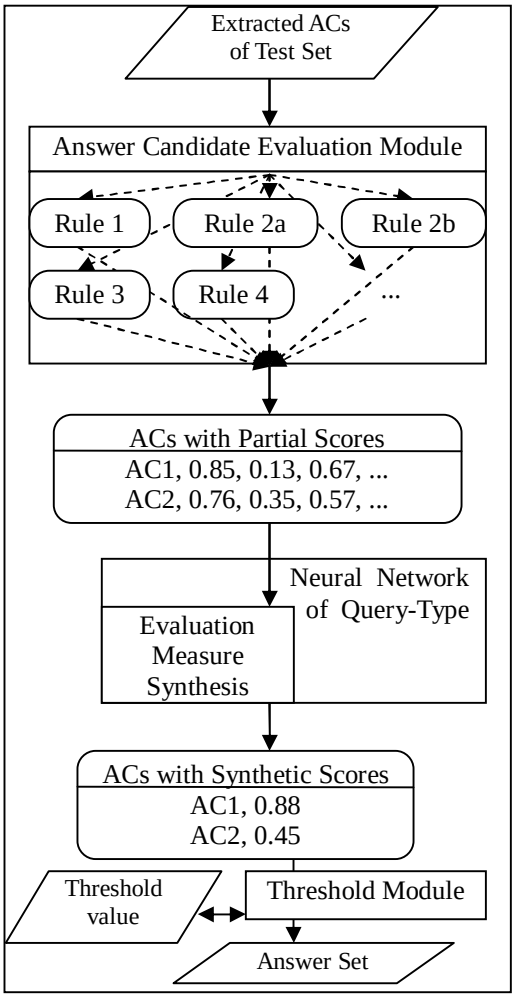

Figure 5. Evaluation measure synthesis

The query is processed in the query analysis module, the document retrieval module, and the answer candidate extraction module. Each of the extracted ACs is evaluated in the answer candidate evaluation module, and the module provides partial scores for them. The partial scores are sent to the neural network that corresponds to the Query-Type of the AC, and the synthetic score is obtained. The threshold module determines whether the AC should be added into the answer

set based on the synthetic score. The answer set is obtained when the system evaluates all of the ACs provided by the answer candidate extraction module.

4. Performance Evaluation

4.1 Conditions of the Evaluation

The QA system implementing the new method (NUT-NN) was compared with the base QA system of the current version (NUT-BASE2) by evaluation using 200 queries of QAC2-task2. They were also compared with the closed evaluation results that use the queries of the QAC3 reference-1 run for both the training set and the test set. The evaluation using the queries of QAC2-task1 was not performed because they are not for the list task. The conditions of the evaluation are as follows:

- The training set was 560 queries of QAC2-task1 and the QAC3 reference-1 run.
- The test set was 200 queries of QAC2-task2, and the correct answer set was the answer list that had been distributed on 2004/11/20 by the QAC task organizer.
- The test set of the closed evaluation includes 360 queries of the QAC3 reference-1 run, and the correct answer set of the closed evaluation was the answer list that had been distributed on 2005/8/18 by the QAC task organizer.

Table 3 shows the number of all elements of the training set corresponding to each Query-Type, which correspond to pattern A described in the Section 3.2.3.

Table 3. Number of training sets of each Query-Type (pattern A)

Interrogative Type	Presence of QF	
	present	absent
who	10144	8667
when	1289	7472
how	104882	41053
how_many	9051	1717
where	174565	164397
where-loc	100134	-
where-org	11364	-
what	309634	144880
what-QF	4287	-

4.2 Evaluation Results

Table 4 shows the results of QAC2-task2 and the QAC3 reference-1 run by the base QA system (NUT-BASE2) and the system using the proposed method (NUT-NN). The values shown as the results of QAC2-task2 are the mean F-measure (MF), and those shown as the results of the QAC3 reference-1 run are the modified mean F-measure (MF1) [3].

Table 4. Results of QAC2-task2 and the QAC3 reference-1 run for NUT-BASE2 and NUT-NN

QA system	QAC2 task2 (MF)	QAC3 ref-1 run (MF1)
NUT-BASE2	0.188	0.0856
NUT-NN	0.198	0.0938

In the evaluation using the QAC2-task2 query set, the proposed system achieved a performance improvement with a value of 0.01 for MF. On the other hand, in the evaluation using the QAC3 reference-1 run query set, the proposed system achieved a performance improvement with a value of 0.082 for MF1.

4.3 Consideration of Training Data

As described in previous sections, a neural network is trained with five different patterns of training data for each Query-Type and the best-trained pattern is selected for evaluation measure synthesis. For the Query-Types that have the nature of narrowing ACs, pattern A is selected. Otherwise, pattern E is selected.

In general, the performances for the former Query-Types are better than those for the latter Query-Types. For example, the MF value of the Query-Type 'who-pr' was improved from 0.264 to 0.308, and the value of 'what-pr' was decreased from 0.113 to 0.108. The distribution of the synthetic scores for the latter Query-Types is thought to have multiple peaks, which implies an overlapped distribution of several sub-Query-Types. Therefore, if more detailed classification can be defined for these Query-Types, then the distinct performance would be improved.

5. Summary and Future Works

To solve the problem of the NUT-BASE QA system in the list task clarified by the results of QAC3, a new method that synthesizes multiple evaluation measures using the layered neural networks was proposed.

The proposed method improved the performance

for the list task.

Further improvement is expected if more detailed Query-Type classification can be achieved. The following criteria can be used for classification methods:

- Whether QF is a compound word
- Whether an AC is the subject or object of the verb that is the main issue of the query sentence.
- Whether ACs are aliases of the subject.

Implementation of these criteria will be attempted in a future study.

The NUT-NN QA system uses the BP-learned neural network for evaluation measure synthesis. However, the decision tree technique can also be applied because some partial scores are binary. Implementation of the proposed technique and its performance comparison will also be performed in the future.

References

[1] A. Takano, S. Nishioka, O. Imaichi, M. Iwayama, Y. Niwa, T. Hisamitsu, M. Fujio, T. Tokunaga, M. Okumura, H. Mochizuki and T. Nomoto. Development of the generic association engine for processing large corpora. (in Japanese) http://geta.ex.nii.ac.jp/

[2] T. Akiba, K. Itou and A. Fujii. Question Answering using "Common Sense" and Utility Maximization Principle. In *Working Notes of the Fourth NTCIR Workshop Meeting Part III: Question Answering Challenge (QAC-2)*, pages 297-303, 2004

[3] T. Kato, J. Fukumoto and F. Masui. An Overview of NTCIR-5 QAC3. In *Proceedings of the Fifth NTCIR Workshop*, 2005.

[4] Y. Matsumoto, H. Kitauchi, T. Yamashita, Y. Hirano, H. Matsuda, K. Takaoka and M. Asahara. *Japanese Morphological Analysis System ChaSen version 2.3.3 Manual.* 2003

[5] F. Masui, S. Suzuki and J. Fukumoto. Named Entity Extraction Tool NExT for Text Processing. In *Proceeding of The Eighth Annual Meeting of The Association for NLP*, pages 176-179, 2002. (in Japanese) http://www.ai.info.mie-u.ac.jp/next/next.html

[6] N. Baba, T. Kojima, and S. Ozawa. *Base and Application of the neural network*. Kyoritsu Publishing, 1994.