AUTOMATIC THESAURUS CONSTRUCTION BY
MACHINE LEARNING FROM RETRIEVAL SESSIONS*

U. GÜNTZER,† G. JÜTTNER,‡ G. SEEGMÜLLER,§ and F. SARRE†
†Department of Computer Science, Technical University of Munich, Postfach 202420, D-8000
München 2, FRG, ‡Institute for Application Oriented Knowledge Processing, Ulm, FRG,
and §Gesellschaft für Mathematik und Datenverarbeitung, Bonn, FRG

(Received 27 May 1988; accepted in final form 15 September 1988)

Abstract—Users of information retrieval systems (IRS) know and use many relationships
between concepts a long time before these find their way into textbooks, printed thesauri,
or classification schemes. We present here an IRS component called TEGEN, which taps
this expertise by automatically drawing conclusions from actual search behavior about
possible thesaurus entries. This is done during an iterative knowledge acquisition pro-
cess: only after explicit or implicit confirmation by other users of the IRS during the
knowledge verification process, the results are incorporated into a thesaurus. TEGEN
is written in PASCAL using a knowledge-based programming method. It uses the rela-
tional database system IMF2 and is implemented at the Technical University of Munich
and at the Leibniz Computer Center of the Bavarian Academy of Sciences.

1. INTRODUCTION

Information retrieval systems (IRS) enable searches to be carried out on a collection of doc-
uments. During this process, the user receives information relevant to the questions that
are of interest to him or her.

The information retrieval (IR) is more successful if various concept relationships can
be incorporated into a search request by means of a thesaurus. Examples of such concept
relationships are: synonyms, related terms, broader and narrower terms, and inflected
forms.

Since the conventional manual methods of thesaurus generation are too costly and the
automatic methods have not produced sufficiently good results to succeed in practice [1–
3], we present in this article a thesaurus-generating system designated TEGEN, which pro-
vides a method by which a thesaurus can largely be learned by a computer without major
effort or expenditure.

TEGEN is implemented on top of a multilingual IRS with automatic (and rather
 crude) indexing, which is used by professionals and graduate students. In such an environ-
ment a thesaurus is particularly helpful, namely:

1. The primitive kind of automatic indexing we use extracts all nonstop words from
titles and abstracts of documents. This means that a paper on artificial intelligence
might be indexed by the term “künstliche Intelligenz” if it happens to be written
in German or by “AI” if only this abbreviation occurred in the title. Furthermore,
a paper on trees might be indexed by “tree” or “trees” (or “arbres” etc.). Without
the use of a thesaurus, recall will be rather low.
2. The thesaurus component is used interactively during query sessions. Thus, the user
has immediate control when modifying the query with the help of the thesaurus.
3. Furthermore, in our environment as in many others gains in recall are much
more at a premium than precision, because professionals are very good at screening
out irrelevant material, but cannot afford to overlook relevant things; therefore,
even if one would use the thesaurus only for enlarging queries (with a correspond-
ning loss in precision) this would be beneficial.

* A shorter version of this article was presented at the conference “User-oriented Content-based Text and
Image Handling” (RIA088), MIT, Boston, MA, March 1988.
To summarize: If the indexing method is rather primitive, then a good thesaurus can make up for its deficiencies and achieves considerably better search results. But normally, if an IRS had adequate resources to create and maintain a good thesaurus, presumably it could also index more intelligently in the first place. The TEGEN system is intended to be a remedy to this dilemma. The intelligence of the user population exhibited during query sessions is tapped to construct a thesaurus reflecting expertise, interests, and even jargon of this population. Of course, it remains to be proven that a really good thesaurus can be generated this way.

2. THE BASIC PRINCIPLES OF THE LEARNING PROCESS

2.1 Learning by analyzing

The learning process used by the TEGEN system is called learning by analyzing. This is a refined variant of learning by observation in which, however, an additional feedback component is used to verify the learning results.

This learning process is based on the assumption that a large number of users of an IRS possess expert knowledge in the field to which their research relates. In carrying out their IR, they adopt certain procedures to find the desired documents contained in the collection of data.

The user searches are evaluated online to acquire information regarding significant concepts and their interrelationships simultaneously with the process of providing information concerning the relevant documents. Thus, the knowledge providing and the knowledge acquisition processes work cooperatively.

TEGEN observes the searches carried out by its users and exploits the users’ expert knowledge. Probable semantic relationships among concepts are extracted online by means of acquisition rules from the syntax of a search request and from the responses of the user to certain reactions of the search system.

At the beginning of the learning process, the thesaurus is almost empty and can give only very limited support to an IRS; its quality, however, increases with the number of qualified users and instances of use.

2.2 Feedback from the user

User searches are evaluated online by TEGEN since feedback from the user is necessary for two reasons:

1. ambiguity of conclusions
2. uncertainty of results.

If the analysis of user searches is ambiguous as regards the conclusions to be drawn, the concept relationship is assigned by the user with the aid of explicit feedback. This means direct feedback from the user. The system asks questions to clarify the situation and expects these questions to be answered unambiguously.

Analysis of the search behavior of a user can lead to the acquisition of uncertain or wrong thesaurus contents. For this reason, the intermediate results of learning may be accepted as final only when their validity has been established by a sufficiently large number of verification processes on the basis of implicit or explicit feedback.

Implicit feedback is feedback from the user without direct questions. Thesaurus contents that require clarification are integrated in the search dialogue of the user at suitable points and the user’s reactions to them are used for verification purposes.

2.3 Architecture of the learning system

Concept relationships, which are recognized as a result of the analysis of a user search, are designated intermediate results (IRES) and are initially assigned an appropriate status. These relationships have to be verified by further users so that their validity can be finally
established. A learning intermediate result becomes a final result (FRES) when the verification process is complete.

Thus, TEGEN consists essentially of two parts: knowledge acquisition and knowledge verification. The knowledge acquisition process derives possible thesaurus contents as intermediate results from the user searches. The knowledge verification process checks these intermediate results and converts them into final results. The architecture of the learning IRS is shown in Fig. 1.

The management of the IRES and the recording of the FRES is carried out by the thesaurus. This contains the relations SYNONYMS, RELATED TERMS, HOMONYMS, BASIC and INFLECTED FORMS, BROADER TERMS and NARROWER TERMS, NEGATIVE LIST and POSITIVE LIST, and USEFUL PREFIXES.

A few remarks concerning these relations shall clarify their semantics. The relation SYNONYMS contains not only synonym pairs in the linguistic sense but also translations and abbreviations, e.g. the pairs (artificial intelligence, AI) and (artificial intelligence, künstliche Intelligenz) would qualify for inclusions into the relation SYNONYMS. For multilingual information systems, the semiautomatic construction of translation tables (for at least those terms that are in actual use) is a great advantage.

The relation POSITIVE LIST contains the controlled vocabulary, i.e. those terms that have been used several times as search terms within queries. At a later stage we intend to use this list for indexing. The unary relation NEGATIVE LIST contains stop words like "is", "of", "the", and "der".

The relation USEFUL PREFIXES needs a special comment. It is designed to help the user when using truncation, e.g. "Datenba." shall denote all terms starting with "Datenba . " like "Datenbank", "Datenbanken", etc. If the user truncates too early, too many documents will be derived; if the user truncates too late, recall will be lost. To find an adequate truncation point, the user goes through a term character-by-character from the left to the right and sends these prefixes one after the other as queries to the system. As an example, we display the results of this method for the term "Datenbanksystem" (this is the German word for "database system").

Fig. 1. Architecture of TEGEN.
As one can see, “Datenban.” is not a useful prefix. If “Datenb.” is not good enough (this depends on the rest of the query), then it will not be helpful to replace it by “Datenban.” The user will go on and eventually use “Datenbanks.” instead. Actually, the user will set the truncation point at those points, where a significant drop in the hit rate takes place or where the semantics changes, like between “Daten.” and “Datenb.”

3. KNOWLEDGE ACQUISITION

Acquisition of the intermediate results takes place by means of acquisition rules, which are represented in the form of production rules such as those frequently used for knowledge representation in expert systems [4–6].

The rules can be classified into three different types, which we present in the following sections. Examples of rules implemented in our system are given.

TEGEN currently contains 29 sets of semantic rules with a total of about 300 production rules that are implemented in PASCAL together with the rule interpreter. Experience gained in using the system will result in further acquisition rules.

3.1 Indirect acquisition without feedback

User searches allow an unambiguous conclusion to be drawn as regards a thesaurus content and this can accordingly be directly acquired without any query being directed at the user.

Insofar as rules of this class are used to investigate a similarity relationship, these cannot be specified more precisely as SYNONYMS, RELATED TERMS, or INFLECTED FORMS without feedback from the user. In this class of rules, therefore, similarity relationships are generally assigned to the relation RELATED TERMS and are specified more closely in a further stage of the process.

Example.

Rule 14: Restriction by means of a new search request
Syntax:
if (a) two or more search terms \( x_i \) in a search request \( X \) are combined by OR
and
(b) \( x_i \) occur as key words
and
(c) the search request \( X \) is combined in a further search request \( Y \) by AND or AND NOT with further search terms \( y_j \)
then all \( x_i \) are accepted in pairs as learning IRES in the RELATED TERMS relation.

Semantics of Rule 14. If the user restricts the combination of search terms by the operators AND or AND NOT with further search terms in a further search request, the search terms of the first search request, which are used to form the union, in general
have a conceptual relationship. The fact that the OR combination receives further conceptual treatment by way of restriction is a strong indication that the terms $x_i$ have not been combined accidentally.

Condition (b) of Rule 14 implies that the search terms are complete and sensible concepts even if truncated.

**Example.**

X: (BAUM OR BAEUME OR TREE OR TREES)  
Y: X AND (SUCHE OR SEARCH)  
produces similarity relationships among:  
BAUM, BAEUME, TREE, TREES.

### 3.2 Indirect acquisition with feedback

In this rule class, the conclusions regarding thesaurus contents are ambiguous; the situation is clarified by explicit feedback. Similarity relationships can be unambiguously assigned to the different thesaurus relations on the basis of queries addressed to the user.

**Example.**

**Rule 25: New attempt in the case of AND combinations**

**Syntax:**

if (a) a search request X consists of $n$ search terms $x_i$ with $n \geq 2$
and
(b) the $x_i$ are combined by AND
and
(c) the search request X is repeated in search request Y but one of the search terms involved, $x_j$, is replaced by the search term $y$
and
(d) the search request Y produces less results than X
and
(e) the user gets the results of Y printed
and
(f) . . . some further premise . . .

then (1) ask the user, whether:

(A) $y$ is a narrower term of $x_j$
(B) $y$ is a synonym or a translation of $x_j$
(C) $x_j$ is an inflected form of $y$
(D) $y$ is an inflected form of $x_j$
(E) an affinity relationship exists between $x_j$ and $y$
(F) none of this applies

(2) in case (A)–(E), the pair $(y, x_j)$ is accepted as IRES in the appropriate thesaurus relation.

**Semantics of Rule 25.** If, after an unsuccessful search attempt in which two or more search terms were combined by AND, a fresh attempt is made and one of the search terms $x_j$ involved is replaced by a search term $y$, a conceptual relationship generally exists between $x_j$ and $y$. If replacing $x_j$ by $y$ produces less results and the user is satisfied with the modification, because the results were printed, then it is rather likely that one has found in $y$ a term narrower than $x_j$ or related to $x_j$. This relationship is classified more precisely by queries addressed to the user.

The precondition $n \geq 2$ is necessary so that it can be seen that question Y is an emendation of question X.

**Example.**

X: STORAGE METHOD AND INFORMATION RETRIEVAL  
Y: INVERTED LIST AND INFORMATION RETRIEVAL  
results, after query to user, in:  
INVERTED LIST is a narrower term of STORAGE METHOD.
Of course, answering the question posed by the system requires some effort by the user. On the other hand, the question gives some hints about other possibilities to emend the query. Maybe the user has replaced \( x_j \) by an inflected form and is reminded—when reading alternative (B)—that \( x_j \) should also have been replaced by a synonym. In spite of the fact that the number of hits appeared satisfactory to the searcher, there could be many other ways to improve the query. In this way, attention is drawn to the fact that thesaurus support is available, i.e. instead of being bothered the user might even be grateful for the interference by the system. In any case, the user can switch off the thesaurus component, e.g. if not ready to answer the questions. But even if the user has no immediate advantage for the actual query by answering the questions of TEGEN, every improvement of the thesaurus will be to the benefit of later searchers, i.e. the colleagues of the present user.

Our first experiences show that users are cooperative, but this acceptance question has to be evaluated carefully along the following lines:

1. Anonymous session history protocols are recorded and shall be evaluated manually. There it can be seen how often users have asked the system for thesaurus help and, if users have left the thesaurus mode during a retrieval session (presumably feeling bothered), how many questions had been asked (and answered) and what was the last question.

   Because we anticipate that TEGEN would usually be used within an organization, the organization could make this kind of cooperation by its members obligatory. We are thinking of trying this method for a limited period of time. Of course, users could react by no longer using the system at all or by answering at random. (Even then the verification procedure would prevent incorporation of random results into the thesaurus.) This appears to be rather unlikely, but has to be verified empirically.

2. Additionally, a questionnaire is designed to find out what the users think of the thesaurus and how taxing they feel the additional effort is. It is very well possible that some types of questions are more bothering than others. Then one might do without these special questions, especially if the corresponding rule was not very productive. This can be measured, because we record for every intermediate and every final learning result the rule number responsible for acquiring (or verifying) this bit of knowledge.

3.3 Direct acquisition

As a further possible method of generating learning intermediate results, the user is asked explicitly about thesaurus contents known to him or her if this can be done within the search dialogue at a point that is appropriate from a semantic point of view and without delaying the user.

For this reason, this form of acquisition is used only if the user has previously indicated that he or she wishes thesaurus support (thesaurus mode).

Example.

Rule 29: Direct acquisition of thesaurus contents

Syntax:

if (a) the thesaurus mode is set
and (b) the user requires concepts that have relationship type \( t \) with a search term \( x \)
then
(1) provide with \( x \) all previously verified concepts of this relationship type \( t \)
(2) carry out verification of not yet finally established intermediate results of \( x \) in respect of this relationship type \( t \)
(3) ask the user for further concepts \( y_j \) that have a relationship of type \( t \) with \( x \)
(4) for all \( y_j \) shown, accept the pair \((x,y_j)\) as intermediate results into the thesaurus relation of this relationship type.
Semantics of Rule 29. By means of this acquisition rule, thesaurus contents are obtained directly from the user by means of questions and are entered into the thesaurus as IRES.

Actions (1) and (2) are important for the verification process and also motivate the user to add to the thesaurus content.

Here the acceptance problem is much less critical than in the last section. The user is asked for further concepts $y_i$ that have a relationship of type $t$ with $x$. The user has to enter these terms anyway and is neither delayed nor bothered, because he or she presumably wants to include them into the actual query.

4. KNOWLEDGE VERIFICATION

4.1 Verification of the intermediate results

In this second phase of the learning process, the knowledge and the behavior of the searchers are once again used to confirm or reject intermediate results and thus to assign them a validity status.

For this purpose, learning attributes are allocated to the intermediate results within the thesaurus database: these learning attributes are initialized during the knowledge acquisition process and updated in the course of verification. These learning attributes include:

- **Learning weight** — indicates the degree of validity of a concept relationship as ascertained so far.
- **Learning status** — indicates the current state of the concept relationship within the learning process.
- **Verification confirmation** — indicates the number of confirmations of an intermediate result that have occurred so far.
- **Verification rejection** — records the number of rejections of an intermediate result.

Finally, the date of entry shows the calendar date on which an IRES was acquired. Since, in addition, the number of the relevant acquisition rule is recorded for each intermediate result, TEGEN even possesses a *metalearning capability*. Once TEGEN has been in use for any adequate period of time, it can be learned by checking this rule number record, which acquisition rules have produced particularly good results.

Abuse of TEGEN is prevented by ensuring that no user is ever involved in the verification of an intermediate result that was acquired by him/herself.

4.2 Variables for controlling the learning process

There are some further quantities that express relationships among the learning attributes and are used for controlling the learning process. The verification frequency $h$ is determined by the sum of the verification confirmations and verification rejections. The learning step $g$ determines the amount by which the learning weight is increased, or reduced, depending on user reactions. The acceptance value $a$ is defined as the difference between the number of verification confirmations and the number of verification rejections; it indicates whether an IRES is being accepted or rejected by a majority of users. Finally, the verification barrier $s$ indicates how high the acceptance value must be for an IRES to be finally accepted.

4.3 Generation of the final results

In the course of the learning process, attempts are made to transform IRES, as far as possible automatically, into FRES so that they are available in the thesaurus.

An intermediate result is converted into an established final result when the verification process has been carried out sufficiently often and thus the verification barrier $s$ has been exceeded.

A preliminary final result is obtained when the period of investigation corresponding to learning period $l$ has elapsed and either the IRES has not been verified sufficiently often
or the users have contrary views as to the validity of a concept relationship. Since in these cases an automatic decision is not possible, manual checking by an external expert is necessary.

4.4 Learning targets and learning parameters

In systems using TEGEN, various learning targets—such as a high learning rate, a large number and high quality of automatically generated FRES, and a small number of external manual checking processes—are laid down; to achieve these targets, several learning parameters can be set.

These learning parameters, which can be influenced from outside to a greater or lesser extent, include the number of users, their degree of expertise, the verification barrier \( s \), and the learning period \( I \). The degree of expertise is determined by the homogeneity of the user group and also by their level of knowledge both in the specialized field (domain expertise) and relating to the search system (retrieval expertise).

Since these learning parameters in part have opposite effects, it is necessary to carry out an experimentally based adaptation. For example, an increase in \( s \) will result in an increase in the quality of the automatically generated FRES since a larger number of users have to participate in the verification of an IRES so that their decision becomes more reliable. However, the learning rate is simultaneously reduced and the number of manual checking processes is increased for the same learning period \( I \).

At present, the value \( s \) is set to 5 in our system and the value \( I \) to 90 days. This means, for example, that a presumed thesaurus relationship must have been confirmed five times more often than it has been rejected before it is finally incorporated into the thesaurus as a final result. However, a preliminary decision is taken in any case after 90 days at the most.

5. OUTLOOK

TEGEN is implemented in PASCAL on a CYBER 990 computer, applying knowledge-based programming methods on the basis of a conventional programming language. By using these programming methods, it has been possible to show that their advantages (i.e., exploratory programming, easy maintenance) are also beneficial if a conventional programming language is used. Additionally, this solves the problems involved in connecting knowledge-based systems to conventional databases and other software systems.

At present, TEGEN is being used within the document retrieval system TUBIBMUE. Online searches on 130,000 documents of the Faculty of Mathematics and Computer Science of the Technical University of Munich are possible [7]. A thesaurus is being generated in accordance with the methods described in this article and the final results are offered to the users to support their searches.

It appears likely that term relationships that have been established by several expert users are in fact meaningful, but this has to be proved by a careful validation, e.g., it has to be validated, whether the methods implemented in fact create good thesaurus relationships, especially whether the verification procedures exclude most invalid intermediate results from incorporation into the thesaurus without being too restrictive.

This validation is planned as follows: as already mentioned in Section 3.2, TEGEN records for every intermediate and every final result the number of the rule responsible for this result; even the complete verification and rejection history for every learning result is recorded. In this way, the quantitative productivity of the different rules can be easily measured. Furthermore, it can automatically test whether the results produced by some rule were controversial during the verification process. Thus, the quality of a rule can be at least partially measured.

Finally, a manual evaluation of a representative part of the thesaurus by a committee is indispensable. However, we are rather optimistic that this committee will draw the same conclusions as the researchers during the feedback phase of the verification process.

The quantity and quality of the results generated depend on the tuning parameters.
explained in Section 4.4. Appropriate values for these parameters can only be fixed empirically.

The TEGEN learning system can be integrated into an IRS with a Boolean search interface at any time. Especially, it is possible (and advisable) to use an existing thesaurus to "prime" the system. It should also be mentioned that this type of computer learning will surely not remain limited to document retrieval systems.

REFERENCES