

Efficient Specialization of Relational Concepts

Technical Report PCG-9

Kurt VanLehn

Departments of Psychology and Computer Science Carnegie-Mellon University Pittsburgh, PA 15217 U.S.A.

1 July 1987

Running Head: Specialization of Relational Concepts



Acknowledgments

This research was supported by the Information Sciences Division. Office of the Naval Research, under contract number N00014-86-K-0678. Reproduction in whole or in part is permitted for any purpose of the United States Government. Approved for public release; distribution unlimited.

N/A N/A N/A N/A THTLE (Include Security Classification) Efficient Specialization of Relational Concepts Efficient Specialization of Relational Concepts E PERSONAL AUTHOR(S) Kurt VanLehn Na. TYPE OF REPORT Ish TIME COVERED FROM 86Sep15 TO 91Sep15 87 July 1 SUPPLEMENTARY NOTATION				REPORT DOCU	MENTATION	PAGE		
SECURITY CLASSFECTION JOWNERADING SCHEDUL JUSTREUTION JUSTREUTION SECURITY CLASSFECTION JUSTREUTION JUSTREUTION JUSTREUTION JOWNERADING SCHEDUL JUSTREUTION JUSTREUTION JUSTREUTION JOWNERADING SCHEDUL JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JUSTREUTION JOHAN	a. REPORT SI	ECURITY CLASS	FICATION		16. RESTRICTIVE	MARKINGS		
BECLASSINGLATION / DOWNGRADING SCHEDUL Approved for public release: Distribution unlimited PERCOMMING ORGANIZATION MEMORY NUMBER(S) S. MONITORNO ORGANIZATION APORT NUMBER(S) BECLASSINGLATION MEMORY CACANIZATION (17 approved) Same as Performing Organization Carnegie-Hellon University (17 approved) Same as Performing Organization Carnegie-Hellon University (17 approved) Same as Performing Organization Departments of Psychology and Computer Science 800 N. Quincy Sc. ANDE OF MUMORY SOMOG SOMAG Sb DIFFECTIVEND ORGANIZATION Sb DIFFECTIVE SCIENCE ANDE OF MUMORY SOMAG SOMAG Sb DIFFECTIVE SCIENCE ANDE OF MUMORY SOMAG SOMAG Sb DIFFECTIVE SCIENCE ANDERSSICH, State and DF Computer Science NOOD14-86-K-0678 Carden Science Science Science NOOD17-5000 ADDRESSICH, State and DF Computer Science NOOD14-86-K-0678 Carden Science Scienc	Unclas	asified	N AUTHORITY	ويعتب والمعادين والمستاد وساحد التكريف	3. DISTRIBUTION	AVAILABILITY	OF REPORT	
PERFORMING ORGANIZATION REPORT NUMBER(S) 5. MONITORING ORGANIZATION REPORT NUMBER(S) PCG-9 5. MONITORING ORGANIZATION REPORT NUMBER(S) Carnegie-Hellon University 60 OFRESTVINCL // ADDRESS (Ch, Same, and ZPCOM) 60 OFRESTVINCL // ADDRESS (Ch, Same, and ZPCOM) 60 OFRESTVINCL // ADDRESS (Ch, Same, and ZPCOM) 70 ADDRESS (Ch, Same, and ZPCOM) Departments of Psychology and Computer Science 800 N. Quincy St. Arlington, VA 22217-5000 Arlington, VA 22217-5000 awawe of FUNDING, FORSORING 80 OFACE SYMBOL // ADDRESS (Ch, Same, and ZPCOM) 80 OFACE SYMBOL // ADDRESS (Ch, Same, and ZPCOM) 80 OFACE SYMBOL // ADDRESS (Ch, Same, and ZPCOM) 80 OFACE SYMBOL // ADDRESS (Ch, Same, and ZPCOM) 80 OFACE SYMBOL // ADDRESS (Ch, Same, and ZPCOM) 80 OFACE SYMBOL // ADDRESS (Ch, Same, and ZPCOM) 80 OFACE SYMBOL // ADDRESS (Ch, Same, and ZPCOM) 80 OFACE SYMBOL // ADDRESS (Ch, Same, and ZPCOM) 80 OFACE SYMBOL // ADDRESS (Ch, Same, and ZPCOM) 80 OFACE SYMBOL // ADDRESS (Ch, Same, and ZPCOM) 80 OFACE SYMBOL // ADDRESS (Ch, Same, and ZPCOM) 80 OFACE SYMBOL <t< td=""><td></td><td></td><td></td><td>··· 6</td><td></td><td></td><td></td><td></td></t<>				··· 6				
PCG-9 Same as Performing Organization a NAME OF PERFORMING (RCANIZATION Carnegia-Mellon University Ex OFFEE SYMMOL (# applicable) 72 NAME OF MONTORING ORGANIZATION Information Sciences Division Office of Naval Research (Code: 1133) c appertments of Psychology and Computer Science Performing NR 15213 Research (Code: 1133) a wake of FUNDING (FROMSORING ORGANIZATION ADDRESS (Chy, Stee, and ZP Code) To appertments of Psychology and Computer Science Antington, VA 22217-5000 a wake of FUNDING (FROMSORING ORGANIZATION Ame as Monitoring Organization Bo OFFICE SYMMOL (# applicable) 9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER NO0014-86-K-0678 a work of Funding (Fromsoring Condensation) Diffect Studyer, NUMBERS PA00005ub201/7-4-86 (# applicable) N/A N/A N/A N/A N/A N/A ADDRESS (Chy, Stee, and ZP Code) To Souther of Funding Numbers Pace Science NO014-86-K-0678 abs for the specialization of Relational Concepts Provide Science Prove Science N/A N/A Thile (Include Security Classification) Exercise Science Souther of Fundical Prove Science N/A N/A Supplet Mental world (S) Kurt Vallehn Subject TERMS (Continue on reverse if necessary and dentity by Book number) Nachine learning, artificial Intelligence, version spaces, concept induction, (Sup), (Sup), (Sup), (Sup), (Sup), (Sup), (Sup), (Sup), (Distribu	tion unlim	ited	
PCC-9 a NAME OF PERFORMING (RCANIZATION Carnegia-Meilon University Bb OFFICE SYMMOL (M applicable) 7a NAME OF MONITORING ORGANIZATION Information Sciences Division Office of Naval Research (Code 1133) Departments of Psychology and Computer Science Pictsburgh, PA 12213 To Domessic(R, Stee, and JP Code) Departments of Psychology and Computer Science Monitoring Organization Be OFFICE SYMMOL (M applicable) 9 POCCURATENT INSTRUMENT IDENTIFICATION NUMBER Processer) auxet of Fundom(SSONMOK) ORGANIZATION ame as Monitoring Organization Be OFFICE SYMMOL (M applicable) 9 POCCURATENT INSTRUMENT IDENTIFICATION NUMBER PROCESSERVE INSTRUMENT IDENTIFICATION NUMBER PROCESSERVE INSTRUMENT IDENTIFICATION NUMBER PROCESSERVE INSTRUMENT IDENTIFICATION NUMBER PRODUCT INSTRUMENT IDENTIFICATION NUMBER PROCESSERVE INSTRUMENT IDENTIFICATION NUMBER PROVE OF REPORT INFORMATION Efficient Specialization of Relational Concepts PRESONAL AUTHOR(S) Rurt VanLehn a TYPE OF REPORT INFORMATION SUPPLEMENTARY NOTATION INFORMATION SubJECT TEAMS (Continue on reverse if necessary and dentity by block number) Machine Learning, artificial intelligence, version spaces, concept induction, SCI., at Nate of REPORT INFORMATION ANALLABULTY OF ABSTRACT Continue on reverse if necessary and identify by block number) DISTRUMITON: AVAILABULTY OF ABSTRACT Paces concept induction, SCI., at Name of REPORTING AND ANALLABULTY OF ABSTRACT (Continue on reverse if necessary and identify by block number) Aname of REPORTING AND ANALLABULTY OF ABSTRACT (202) OFFICE SYMEOL (202) OFFICE SYMEOL	PERFORMIN	NG ORGANIZAT	ION REPORT NUMBE	ER(S)				
Carnegie-Mellon University (# appicable) Information Sciences Division Office of Naval Research (Code 1133) Carnegie-Mellon University To ADDRSS (GN, Stan, and ZP Code) To ADDRSS (GN, Stan, and ZP Code) Departments of Psychology and Computer Science 800 N. Quincy St. Arlington, VA 22217-5000 Nonderstrong Soponsoland ORGANIZATION Sto Office of Naval Research (Code 1133) Ame of Submids SPONSORING ORGANIZATION Sto Office of Submids VA 22217-5000 Ame of Submids SponsoRand ORGANIZATION Sto Office of Submids VA 22217-5000 Ame of Submids SponsoRand ORGANIZATION Sto Office of Submids VA 22217-5000 Ame of Submids SponsoRand ORGANIZATION Sto Office of Standerson Submids VA 22217-5000 Infle (Include Security Organization (f sponsore) Sto Office of Submids VA 22217-5000 Infle (Include Security Organization (f sponsore) Sto Office of Store Store (GN VA VALABULTY OF ASSTRACT Store (GN VALABULTY OF ASSTRACT Store (GN VALABULTY OF ASSTRACT Store (GN VALABULTY OF ASSTRACT Store (GN VALABULTY OF ASSTRACT Store) Inflet (Include Security Of Asstract Store) Store of Store (GN VALABULTY OF ASSTRACT Store) Inflet (Include Security Of Asstract Store) Store of Store) See reverse side. See reverse side. See reverse side. See reverse Store) Inflet (Include Acos Noffice) Store (Group Costre)	PCG-9				Same as	reriorming	organizati	.on
Carnegie-Hellon University Information Sciences by Using Office of Naval Research (Code 1133) C ADDRESS (CN, Stam, and 2P Code) Departments of Psychology and Computer Science B00 N. Quincy St. Office of Naval Research (Code 1133) A Monto Sponsonna OBE OFFICE SYMBOL ORGANIZATION An and Of SubDiver Sponsonna Bb OFFICE SYMBOL OF SubDiver Sponsonna Construction Structure Constructure Constructure Constructure SubDiver Stamma Constructure	. NAME OF	PERFORMING	REANIZATION					
ADDRESS (GP, State, and ZW Code) Departments of Psychology and Computer Science BOD N. Quincy St. Arington, VA 22217-5000 ADDRESS (GP, State, and ZW Code) Bed Stranger A 15213 Supercent NestRument Dentification Number Organization Supercent NestRument Dentification Number No0014-86-K-0678 DODRESS (GP, State, and ZW Code) Departments of Psychology and Computer Science No0014-86-K-0678 NO0014-86-K-0678 DODRESS (GP, State, and ZW Code) Department Dentification Number No0014-86-K-0678 NO0014 N/A N/	Carneg	gie-Mellon	University					
Pittsburgh, PA 15213 Arlington, VA 22217-5000 Name OF FUNDING: SPONSORING ORGANIZATION Sb OFFICE SYMBOL (If spokesber) 9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N00014-86-K-0678 ame as Monitoring Organization Sb OFFICE SYMBOL (If spokesber) 9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N00014-86-K-0678 ame as Monitoring Organization Sb OFFICE SYMBOL (If spokesber) 9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N00014-86-K-0678 Thue and the Code Machine Instrument IDENTIFICATION NUMBER NOA N/A N/A Thue diverse and the Code Machine Instrument IDENTIFICATION NUMBER NAME OF SECURITY Classification Machine Instrument IDENTIFICATION NUMBER NAME OF SECURITY Classification N/A Thue diverse and the Code reconstruction Kurt VanLehn N/A N/A N/A The of SECONAL AUTHOR(S) Kurt VanLehn Is DuBit Of Seconstruction Is OBJECT TERMS (Continue on reverse if necessary and identity by block number) Subscription Statistic Codes Is SubJECT TERMS (Continue on reverse if necessary and identity by block number) ABSTRACT (Continue on reverse if necessary and identity by block number) See reverse side. See reverse side. See reverse side. Distribution (Authing ID) Is Subject TERMS (Continue on reverse if necessary and identity by block number)	ADDRESS	(City, State, and	d ZIP Code)					
ORGANIZATION (# appicable) N00014-86-K-0678 Ime as Monitoring Organization 0 SOURCE OF FUNDING NUMBERS \$400005ub201/7-4-86 ADDRESS(Cir), State, and ZiP Code) PROGRAM N/A ITTLE (Include Security Classification) Element No N/A N/A Efficient Specialization of Relational Concepts N/A N/A N/A PERSONAL AUTHOR(S) Kurt VanLehn N/A N/A N/A a TYPE OF REPORT [13h TIME COVERED 14 Date OF REPORT, Year, Month, Day) 15 PAGE COUNT suppleventrary votation (************************************				Computer Scien			217-5000	
ame as Monitoring Organization N00014-86-K-0678 : ADDAESS(Ciry, State, and SIP Code) 10. SQURCE OF FUNCING NUMBERS \$400005Ub201/7-4-86 PEGGORAM PEGLECT NOK N/A N/A N/A N/A Superstructure N/A ABSTRACT CODES Image: Superstructure See reverse side. See reverse side. See reverse side. See reverse side. NAME OF REPORT REPORT CODES Same as RPT Darc Uses NAME OF REPORT REPORT RE			NSORING		9 PROCUREMEN	TINSTRUMENT	DENTIFICATION	NUMBER
ADDRESS (Ciry, State, and ZP Code) III SOURCE OF FUNDING NUMBERS p400005Ub201/7-4-86 PROGRAM PROGRAM ELEMENT NO N/A IASURCE OF FUNDING NUMBERS p400005Ub201/7-4-86 PROGRAM ILEMENT NO N/A N/A TITLE (Include Security Classification) Efficient Specialization of Relational Concepts PERSONAL AUTHOR(S) Kurt VanLehn a TYPE OF REPORT 13b TIME COVERED SUPPLEMENTARY NOTATION If reverse of report 13b TIME COVERED SUPPLEMENTARY NOTATION If reverse of local Subject TERMS (Continue on reverse if necessary and identity by block number) Image: Subject Continue on reverse if necessary and identity by block number) ABSTRACT (Continue on reverse if necessary and identity by block number) See reverse side.	•		Organization		N00014-86	-K-0678		
COSATI CODES 11 DATE (Include Security Classification) Efficient Specialization of Relational Concepts E PERSONAL AUTHOR(S) Kurt VanLehn ta TYPE OF REPORT 13 TIME COVERED Fechnical 14 Dare OF REPORT 13 TIME COVERED 14 Dare OF REPORT 13 TIME COVERED 14 Dare OF REPORT 13 TIME COVERED 14 Dare OF REPORT 15 TIME COVERED 14 Dare OF REPORT 15 TIME COVERED 16 TYPE OF REPORT 17 TYPE OF REPORT 18 TUBE TIME COVERED 18 TUBE TIME COVERED 18 TUBE TO 91 Sep15 14 Dare OF REPORT 19 TIME TO 7 AVAILABLET 10 DATE OF REPORT 19 TIME TO 7 AVAILABLET 10 DISTRIBUTION / AVAILABLETY OF ARSTRACT	and the second		and the second	<u> </u>	10 SOURCE OF	-UNDING NUM	BERS p400005	ub201/7-4-86
N/A N/A N/A N/A N/A ITTLE (Include Security Classification) Efficient Specialization of Relational Concepts Efficient Specialization of Relational Concepts Image: Personal author(s) Kurt VanLehn Is TIME COVERED Fechnical Is TIME COVERED From 865ep15 SuppleMentary Notation From Costin Codes Field GROUP SuppleMentary Notation Field GROUP Supplementary Notation Field GROUP Supplementary Notation Machine Learning, artificial intelligence, version spaces, concept induction, (Strategrameter) Asstract (Continue on reverse if necessary and identify by block number) See reverse side. See reverse side. Supplementary Strategrameters in Security Classification Supplemetary Strategrametery Strat								WORK UNIT
TITLE (Include Security Classification) Efficient Specialization of Relational Concepts PERSONAL AUTHOR(S) Kurt VanLehn Is TYPE OF REPORT Tab TIME COVERED Fachnical THE (Include Security Classification) Supple VENTARY NOTATION COSATI CODES COSATI CODES Value Value Supple VENTARY NOTATION COSATI CODES Value Value Supple VENTARY NOTATION COSATI CODES Value Value Value Value Supple VENTARY NOTATION COSATI CODES Value Value Value ABSTRACT (Continue on reverse if necessary and identify by block number) See reverse side. See reverse side. See reverse side. Value See Seconsplit - Concusters Value Sastract Seconsplit - Concusters Value Sastract Seconsplit - Concusters Value Sastract Seconsplit - Concusters Valu							N/A	1
PIELD GROUP Sub-GROUP Machine learning, artificial intelligence, version spaces, concept induction, (Sub, art) ABSTRACT (Continue on reverse if necessary and identify by block number) See reverse side. See reverse side. Unctassified/unuimited Same of RESPONSIBLE INDIVIDUAL Alan Meyrowitz	Echnica	REPORT	135 TIME CO	OVERED	14 DATE OF REPO 87 July	ORT (Year, Mon 1	th, Day) 15. PA	GE COUNT
Distribution/ Availability of abstract	a TYPE OF Technica	REPORT	135 TIME CO	OVERED Sep15 to 91Sep15			th, Oayj 15. Pa 7	AGE COUNT
ABSTRACT (Continue on reverse if necessary and identify by block number) See reverse side. 21 ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED/UNLIMITED SAME AS RPT DTIC USERS NAME OF RESPONSIBLE INDIVIDUAL Alan Meyrowitz 220 690-4302 220, 000-4302	a. TYPE OF Technica: SUPPLEME	REPORT	135 TIME CO FROM <u>86</u> ION	OVERED Sep15 TO 91 Sep15	SC SIJC -	se if necessary	and identify by	block number)
See reverse side. Distribution / Availability of Abstract Dunclassified/unlimited Same as RPT DITIC USERS NAME OF RESPONSIBLE INDIVIDUAL Alan Meyrowitz 220, 75, 590-4302 Area Code) 222, OFFICE SYMBOL 1143	a. TYPE OF Eechnica: SUPPLEME	REPORT	135 TIME CO FROM <u>86</u> ION	OVERED Sep15 TO 91Sep15 fr YCVEV (18 SUBJECT TERMS (Machine lear	Continue on reven rning, artifi	e if necessary icial inte	and identity by lligence, v	block number)
DUNCLASSIFIED/UNLIMITED D SAME AS RPT DOTIC USERS a NAME OF RESPONSIBLE INDIVIDUAL Alan Meyrowitz 220 (202) 696-4302 220 (143)	A TYPE OF Eechnica SUPPLEME	REPORT	136 TIME CO FROM <u>86</u> ION CODES SUB-GROUP	OVERED Sep15 TO 91Sep15 fr YCVEV (18 SUBJECT TERMS (Machine lean spaces, cond	Continue on reven rning, artificept induction	e if necessary icial inte	and identity by lligence, v	block number)
Alan Meyrowitz (202) 696-4302	Ra. TYPE OF Technica: SUPPLEME! FIELD	REPORT	136 TIME CO FROM <u>86</u> ION CODES SUB-GROUP	OVERED Sep15 TO 91Sep15 (18 SUBJECT TERMS (Machine lean spaces, cond and identify by block i	GC GILC - Continue on rever rning, artificept induction number)	e if necessary icial inte	and identity by lligence, v	block number)
	ABSTRACT	REPORT	I 36 TIME CO FROM 86	OVERED Sep15 TO 91Sep15 (18 SUBJECT TERMS (Machine lean spaces, cond and identify by block i See ret	Continue on reverse rning, artific cept induction number) verse side.	e if necessary icial inte on, (Shir); ECURITY CLASSI	and identity by lligence, V	block number) rersion

Abstract

1

This research note presents an algorithm for a common induction problem, the specialization of overly general concepts. A concept is too general when it matches a negative example. The particular case addressed here assumes that concepts are represented as conjunctions of predicates, that specialization is performed by conjoining predicates to the overly general concept, and that the resulting specializations are to be as general as possible. It is shown that the problem is simple if the concept representation language is propositional, but NP-complete if the language is first-order (i.e., relational). Nonetheless, there exists an algorithm, based on manipulation of bit vectors, that provides good average-case performance.

Accession For NTIS GRA&I DTIC TAB Unannounced Justification By_ Distribution/ Artility Codes and/or Special <u>5:5</u>t

DTIC COPY This research note presents an adequately efficient algorithm for a common type of induction problem, the specialization of overly general concepts. A concept is too general when it matches a negative example. This problem is fairly common in inductive concept formation. For instance, when updating the G set of a version space (Mitchell, 1982), if the current example is negative and some of the elements of the G set match it, then the first step of the update-G process is to specialize those concepts. In algorithms, such as ID3 (Quinlan, 1986), that represent concepts as decision trees, extending a node of the tree is specialization of the concept expressed by the branch of the tree ascending from that node to the root.

This note concerns the special case wherein concepts are represented as a conjunction of atomic predicates. Much work in concept formation has used such conjunctive representations. It is also assumed that specialization is limited to conjoining predicates to the overly general concept. Although there are several other ways to specialize concepts, this is one of the most common ones. For instance, it is the only form of specialization allowed in ID3. Lastly, it is assumed that the induction algorithm should produce concepts that are as general as possible. Again, this is a common assumption in machine learning work.

If the representation is further restricted so that the predicates have no arguments, then the representation is equivalent to a feature set or an attribute-value vector. For this propositional case, there are several well-know algorithms for solving the specialization problem. An easy one to describe is the problem of updating a version space where both concepts and examples are represented as sets of features, and matching is implemented by set inclusion. The algorithm takes three arguments: the overly general concept, C; the negative example, N; and a third concept, P, which is a member of the S set of the version space. The algorithm simply takes each feature in the set difference P-N, and creates a new concept by adding it to C. The algorithm returns this set of concepts; each of which specializes C (i.e., it is a superset of C) and fails to match N (i.e., it is not a subset of N). The basic idea of this algorithm appears in the other propositional algorithms as well, with the modification that the source, P, of positive features may not be a member of the S set but may instead be a positive example (as in Langley's (1987) discrimination algorithm) or even the whole vocabulary of predicates (as in ID3).

The basic idea of the proposition algorithms is to find a set of differences between P and N, and add each to C. Because the new concepts are required to be as general as possible, only

"minimal" differences between P and N are added to C. If the representation is propositional, then minimal differences are always exactly one predicate, and never a conjunction of predicates. However, when the algorithm is extended to relational representations (i.e., conjunctions of predicates with variables) a minimal difference between P and N is not necessarily a single relation, but in some cases can be a small conjunction of relations. For instance, consider the following situation, where numbers represent objects and letters are variables:

 $N = P(1,2) \in Q(2,1)$ $S = P(y,x) \in Q(x,y) \in P(x,y)$

The minimal differences between P and N are subconcepts of P that do not match N. In this case, there are seven such subconcepts, and only three fail to match N:

1.	P(y,x)				Matches	under	the	substitution	{ x /2,	y/1}
2.	$Q(\mathbf{x}, \mathbf{y})$				Matches	under	the	substitution	{ x /2,	y/1
3.	P(x,y)				Matches	under	the	substitution	${x/1},$	y/2}
4.	₽(y,x)	4	$Q(\mathbf{x}, \mathbf{y})$		Matches	under	the	substitution	{ x /2,	y/1}
5.	P(y, x)	£	P(x,y)			No		tch		-
6.	P(x,y)	6	$Q(\mathbf{x}, \mathbf{y})$			No		tch		
7.	P (y, x)	6	$Q(\mathbf{x},\mathbf{y})$	٤	P(x,y)	No	o ma	tch.		

Given some overly general concept C that matches N, any of the last three could be conjoined to C in order to specialize it. However, the last subconcept is not a minimal difference, because it properly includes other subconcepts that do not match N (i.e., subconcepts 5 and 6), so only subconcepts 5 and 6 should be added to C. This produces two new maximally general concepts, assuming that C was maximally general before N was received. The main point of this example is that the smallest concepts that serve to specialize C are conjunctions of two predicates.

Incidentally, conjoining a subconcept to C can be nontrivial. For instance, if C is $P(u, v) \in Q(v, u)$, then conjoining $P(x, y) \in Q(x, y)$ to it should yield both $P(u, v) \in Q(v, u) \in Q(u, v)$ and $P(u, v) \in P(v, u) \in Q(v, u)$. Fortunately, this problem is similar to the Update-S problem, and nearly the same algorithm can be used for it.

A direct, but inefficient way to generate minimal differences between P and N is to search for them by chaining together predicates that share variables. At one time, the procedure-learning program Sierra (VanLehn, 1987) used this technique for updating its version spaces. It often took 30 hours or more just to handle a few negative examples. With some help from Johan de Kleer, a much faster algorithm was invented. Now Sierra completes the same calculations in a minute or two.

The key idea is to convert the problem of finding minimal differences into a set covering problem. The first step is to enumerate all substitutions of objects in N for variables in P. For example, if there were 5 variables in P and 7 objects in N, then there are either 7⁵ possible substitutions when distinct variables can be bound to the same object, or 7*6*5*4*3 substitutions when distinct variables must be bound to distinct objects.¹ Having enumerated the substitutions, each predicate in P is assigned a bit vector. The bit vector has one bit for each substitution. If the predicate is in N under a given substitution, then the bit is zero. If the predicate is not in N, then the bit is one. If any predicate has a bit vector that is all ones, then it is not in N under any substitution, so it is a minimal difference between P and N. Assuming there are no such predicates, the algorithm must to find a small set of predicates such that the union of their bit vectors is all ones.

Attaching bit vectors to predicates converts the minimal difference problem into a well-known problem, the set covering problem: Given a target set and a collection of subsets of it, find a cover for the target set, where a *cover* is a set of subsets such that the union of those subsets equals the target set. There are several different versions of the set covering problems, depending on the kind of cover desired. An *irredundant* cover is a cover that is not properly included in any other cover (i.e., none of its subsets is redundant in that it can be removed from the cover without affecting the cover's equality to the target set). Irredundant covers correspond to minimal differences between P and N, which in turn are the differences that lead to maximally general specializations of C. So for most applications, the covering algorithm desired will be one that finds irredundant covers. Another possibility is an algorithm for calculating minimal covers, which are covers with the fewest number of elements. This would cause the specialization algorithm to generate some, but not necessarily all, of the maximally general concepts.

The most straightforward way to generate irredundant covers is breadth-first search. However, this approach is very space inefficient. Sierra uses an algorithm from Wells (1971, section 6.4.3), which is based on depth-first search. The trick is to prune the search whenever adding a new subset to the cover causes the cover to become redundant. Well's algorithm is presented in table 1. The second clause of the Cond implements the search pruning.

¹Sierra's concept representation enforces additional constraints that reduces the number of substitutions still further. This reduction is crucial, because Sierra's concepts usually had between 10 and 50 variables. In retrospect, the same reduction could be achieved more elegantly by assigning types to variables and objects, then enumerating only substitutions that paired objects and variables of the same type.

 Table 1: An algorithm for finding irredundant covers
 (Defun FindCover (Cover Covered Duplicates Usable) (And Usable (Let ((Candidate (Car Usable)) (NewCover (Cons Candidate Cover)) (NewCovered (Union Covered (BitVector Candidate))) (NewDuplicates (Union Duplicates (Intersect Covered (BitVector Candidate))) (NewUsable (Cdr Usable)) (Cond ((For X in NewCover there is (SubsetP (BitVector X) NewDuplicates)) (FindCover Covered Duplicates NewUsable)) ((TotalSetP NewCovered) (Cons NewCover (FindCover Covered Duplicates NewUsable)) (T (Append (FindCover NewCovered NewDuplicates NewUsable) (FindCover Covered Duplicates NewUsable))))))))

FindCover returns a list of irredundant covers. Cover is a list of predicates paired with their bit vectors. Covered is the bit vector for the current coverage of the cover. Duplicates is the bit vector for the substitutions that are covered twice by members of the cover. Usable is a list of potential candidates for adding to the cover.

Table 2 presents estimates for the time and space complexity of this depth-first algorithm, the breadth-first algorithm, and the algorithm based on chaining through variables. Note that all the algorithms are exponential. This is inevitable, because set covering is an NP-complete problem (Aho, Hopcroft & Ullman, 1974, theorem 10.2.9). However, the exponents tends to be small in typical applications. For Sierra, p and n are usually about 100, and m is usually 1 or 2.

Table 2: Complexity estimates for three minimal difference algorithms

Algoritrim	Time	Space
Depth-first	(p log p) ^m	p log p
Breadth-first	þ	p ^m
Chaining	p ^m n ^v	pm

Where: p is the number of predicates in P, n is the number of predicates in N, m is the average number of predicates in a minimal difference, and v is the average number of variables in a minimal difference.

As mentioned earlier, this algorithm was used with success as part of a version space maintenance module. It would be equally useful in machine learning programs, such as AQ11 (Michalski & Larson, 1978), that employ version-space-like techniques as components. It would probably be useful for programs, such as ID3 (Quinlan, 1986) and PRISM (Langley, 1987), that induce decision trees. All these programs were initially developed with propositional concept representations. The algorithm described above extends them for use with relational concept representations.

References

- Aho, A.V., Hopcroft, J.E. & Ullman, J.D. (1974). The Design and Analysis of Computer Algorithms. Reading, MA: Addison-Wesley.
- Langley, P. (1987). A general theory of discrimination learning. In Klahr, D., Langley, P. & Neches, R. (Ed.), *Production System Models of Learning and Development*. Cambridge, MA: MIT Press.
- Michalski, R.S. & Larson, J.B. (1978). Selection of most representative training examples and incremental generation of VL1 hypotheses: the underlying methodology and the description of programs ESEL and AQ11 (Technical Report 867). Computer Science Department, University of Illinois.

Mitchell, T.M. (1982). Generalization as search. Artificial Intelligence, 18, 203-226.

Quinlan, J. R. (1986). The effect of noise on concept learning. In R. S. Michalski, J. G. Carbonell,
& T. M. Mitchell (Ed.), *Machine Learning: An Artificial Intelligence Approach. Volume II.* Los Altos, CA: Morgan Kaufman.

VanLehn, K. (1987). Learning one subprocedure per lesson. *Artificial Intelligence*, *31*(1), 1-40. Wells, M.B. (1971). *Elements of Combinatorial Computing*. New York, NY: Pergamon Press.