Technical Report: A Systematic Mapping Study of Machine Learning for Software Traceability

1. Extracted Data

1.1 Extracted Data for RQ1

Index	Title	Author	Subject	Year	Venue	Publication type
S1	An extended knowledge representation learning approach for context-based traceability link recovery	Guoshuai Zhao Tong Li Zhen Yang	Researcher	2020	International Conference on Software Engineering and Knowledge Engineering (SEKE)	Conference
S2	An Improved Approach to Traceability Recovery Based on Word Embeddings	Teng Zhao Qinghua Cao Qing Sun	Student	2017	Asia-Pacific Software Engineering Conference (APSEC)	Conference
S3	An information theoretic approach for extracting and tracing non-functional requirements	Anas Mahmoud	Researcher	2015	International Requirements Engineering Conference (RE Conference)	Conference
S4	Application of reinforcement learning to requirements engineering requirements tracing	Hakim Sultanov Jane Huffman Hayes	Researcher	2013	International Requirements Engineering Conference (RE Conference)	Conference
S5	ATLaS: A Framework for Traceability Links Recovery Combining Information Retrieval and Semi-Supervised Techniques	Emma Effa Bella Stephen Creff Marie-Pierre Gervais Reda Bendraou	Student	2019	The Enterprise Computing Conference (EDOC)	Conference
S6	Traceability recovery between bug reports and test cases-a Mozilla Firefox case study	Guilherme Gadelha Franklin Ramalho Tiago Massoni	Researcher	2021	Automated Software Engineering (ASE)	Journal
S7	Automatic traceability link recovery via active learning	Tianbao Du Guohua Shen Zhiqiu Huang Yaoshen Yu Dexiang Wu	Student	2020	Frontiers of Information Technology & Electronic Engineering (FRONT INFORM TECH EL)	Journal
S8	Automatic Traceability Maintenance via Machine Learning Classification	Chris Mills Javier Escobar-Avila Sonia Haiduc	Researcher	2018	International Conference on Software Maintenance and Evolution (ICSME)	Conference
S9	Automating traceability link recovery through classification	Chris Mills	Researcher	2017	European Software Engineering	Conference

					Conference (ESEC)	
S10	Clustering for Traceability Managing in System Specifications	Manel Mezghani Juyeon Kang Eun-Bee Kang Florence Sedes	Researcher	2019	International Requirements Engineering Conference (RE Conference)	Conference
S11	Combining Machine Learning and Logical Reasoning to Improve Requirements Traceability Recovery	Tong Li Shiheng Wang David Lillis Zhen Yang	Researcher	2020	Applied Sciences (APPS)	Journal
S12	Detecting, classifying, and tracing non-functional software requirements	Anas Mahmoud Grant Williams	Researcher	2016	Requirements Engineering (RE)	Journal
S13	Enhancing Automated Requirements Traceability by Resolving Polysemy	Wentao Wang Nan Niu Hui Liu Zhendong Niu	Researcher	2018	International Requirements Engineering Conference (RE Conference)	Conference
S14	A Machine Learning Approach for Determining the Validity of Traceability Links	Chris Mills Sonia Haiduc	Researcher	2017	International Conference on Software Engineering (ICSE)	Conference
S15	Enhancing Unsupervised Requirements Traceability with Sequential Semantics	Lei Chen Dandan Wang Junjie Wang Qing Wang	Researcher	2019	Asia-Pacific Software Engineering Conference (APSEC)	Conference
S16	Estimating the number of remaining links in traceability recovery	Davide Falessi Massimiliano Di Penta Gerardo Canfora Giovanni Cantone	Researcher	2017	Empirical Software Engineering (ESE)	Journal
S17	Evaluation of Textual Similarity Techniques in Code Level Traceability	Viktor Csuvik Andras Kicsi Laszlo Vidacs	Student	2019	International Conference on Computational Science and Its Applications (ICCSA)	Conference
S18	Improving the effectiveness of traceability link recovery using hierarchical bayesian networks	Kevin Moran David N. Palacio Carlos Bernal-Cardenas Daniel McCrystal Denys Poshyvanyk Chris Shenefiel Jeff Johnson	Practitioner	2020	International Conference on Software Engineering (ICSE)	Conference
S19	Traceability Link Recovery between Requirements and Models using an Evolutionary Algorithm Guided by a Learning to Rank Algorithm: Train control and management case	Ana C. Marcen Raul Lapena Oscar Pastor Carlos Cetina	Researcher	2020	Journal of Systems and Software (JSS)	Journal
S20	Information retrieval versus deep learning approaches for generating traceability links in bilingual projects	Jinfeng Lin Yalin Liu Jane Cleland-Huang	Researcher	2022	Empirical Software Engineering (ESE)	Journal

S21	Issue Link Label Recovery and Prediction for Open Source Software	Alexander Nicholson Jin L.C. Guo	Student	2021	International Requirements Engineering Conference (RE Conference)	Conference Workshop
S22	Large Scale Evaluation of Natural Language Processing Based Test-to-Code Traceability Approaches	Andras Kicsi Viktor Csuvik Laszlo Vidacs	Researcher	2021	IEEE Access (IEEE ACCESS)	Journal
S23	Leveraging Historical Associations between Requirements and Source Code to Identify Impacted Classes	Davide Falessi Justin Roll Jin L.C. Guo Jane Cleland-Huang	Researcher	2020	IEEE Transactions on Software Engineering (IEEE T SOFTWARE ENG)	Journal
S24	On the effect of incompleteness to check requirement-to-method traces	Mouna Hammoudi Christoph Mayr-Dorn Atif Mashkoor Alexander Egyed	Researcher	2021	ACM Symposium On Applied Computing (SAC)	Conference
S25	On the relationship between similar requirements and similar software	Muhammad Abbas Alessio Ferrari Anas Shatnawi Eduard Enoiu Mehrdad Saadatmand Daniel Sundmark	Practitioner	2022	Requirements Engineering (RE)	Journal
S26	Tracing with Less Data: Active Learning for Classification-Based Traceability Link Recovery	Chris Mills Javier Escobar-Avila Aditya Bhattacharya Grigoriy Kondyukov Shayok Chakraborty Sonia Haiduc	Researcher	2019	International Conference on Software Maintenance and Evolution (ICSME)	Conference
S27	Semantically Enhanced Software Traceability Using Deep Learning Techniques	Jin L.C. Guo Jinghui Cheng Jane Cleland-Huang	Student	2017	International Conference on Software Engineering (ICSE)	Conference
S28	Semi-Automated Feature Traceability with Embedded Annotations	Hadil Abukwaik Andreas Burger Berima Kweku Andam Thorsten Berger	Researcher	2018	International Conference on Software Maintenance and Evolution (ICSME)	Conference
S29	Source Code Level Word Embeddings in Aiding Semantic Test-to-Code Traceability	Viktor Csuvik Andras Kicsi Laszlo Vidacs	Student	2019	ICSE Workshop on Software and Systems Traceability (SST)	Conference Workshop
S30	Tackling the term-mismatch problem in automated trace retrieval	Jin L.C. Guo Marek Gibiec Jane Cleland-Huang	Student	2017	Empirical Software Engineering (ESE)	Journal
S31	TCTracer: Establishing test-to-code traceability links using dynamic and static techniques	Robert White Jens Krinke	Practitioner	2022	Empirical Software Engineering (ESE)	Journal
S32	Toward accurate link between code and software documentation	Yingkui Cao Yanzhen Zou	Researcher	2018	Science China Information Sciences	Journal

		Yuxiang Luo Bing Xie Junfeng Zhao			(SCIS)	
S33	Towards feature-aware retrieval of refinement traces	Patrick Rempel Patrick Mader Tobias Kuschke	Student	2013	ICSE Workshop on Traceability in Emerging Forms of Software Engineering (TEFSE)	Conference Workshop
S34	Traceability Transformed_ Generating more Accurate Links with Pre-Trained BERT Models	Jinfeng Lin Yalin Liu Qingkai Zeng Meng Jiang Jane Cleland-Huang	Researcher	2021	International Conference on Software Engineering (ICSE)	Conference
S35	Towards the automatic classification of traceability links	Chris Mills	Researcher	2017	International Conference on Automated Software Engineering (ASE Conference)	Conference
S36	Tracing Requirements as a Problem of Machine Learning	Zeheng Li LiGuo Huang	Student	2018	International Journal of Software Engineering & Applications (IJSEA)	Journal
S37	Traceability in the wild: automatically augmenting incomplete trace links	Michael Rath Jacob Rendall Jin L.C. Guo Jane Cleland-Huang Patrick Mader	Researcher	2018	International Conference on Software Engineering (ICSE)	Conference

1.2 Extracted Data for RQ2

Index	Title	ML Models	Stage
S1	An extended knowledge representation learning approach for context-based traceability link recovery	Decision Tree (DT) GBDT Naive Bayes (NB) SVM	link generation stage
S2	An Improved Approach to Traceability Recovery Based on Word Embeddings	Word2vec Ranking SVM	preprocessing stage link generation stage
S3	An information theoretic approach for extracting and tracing non-functional requirements	Hierarchical Agglomerative Clustering (HAC)	preprocessing stage
S4	Application of reinforcement learning to requirements engineering requirements tracing	Reinforcement Learning	link generation stage
S5	ATLaS: A Framework for Traceability Links Recovery Combining Information Retrieval and Semi-Supervised Techniques	Word2vec GloVe Label spreading	preprocessing stage link generation stage
S6	Traceability recovery between bug reports and test cases-a Mozilla Firefox case study	GloVe	preprocessing stage
S7	Automatic traceability link recovery via	Active Learning	link generation stage

	active learning		
S8	Automatic Traceability Maintenance via Machine Learning Classification	K nearest neighbors (KNN) Naive Bayes (NB) Logistic Regression (LR) SVM Random Forest (RF)	link generation stage
S9	Automating traceability link recovery through classification	Decision Tree (DT) Random Forest (RF) K nearest neighbors (KNN) Naive Bayes (NB)	link generation stage
S10	Clustering for Traceability Managing in System Specifications	K-means	link generation stage
S11	Combining Machine Learning and Logical Reasoning to Improve Requirements Traceability Recovery	Doc2vec Decision Tree (DT) K nearest neighbors (KNN) Random Forest (RF) GBDT	preprocessing stage link generation stage
S12	Detecting, classifying, and tracing non-functional software requirements	Hierarchical Agglomerative Clustering (HAC) K-medoids	preprocessing stage
S13	Enhancing Automated Requirements Traceability by Resolving Polysemy	FNN Word2vec	preprocessing stage
S14	A Machine Learning Approach for Determining the Validity of Traceability Links	Random Forest (RF)	link generation stage
S15	Enhancing Unsupervised Requirements Traceability with Sequential Semantics	Word2vec Doc2vec	preprocessing stage
S16	Estimating the number of remaining links in traceability recovery	Decision Tree (DT) Bagging K nearest neighbors (KNN) Logit Boost Naive Bayes (NB)	link generation stage
S17	Evaluation of Textual Similarity Techniques in Code Level Traceability	Doc2vec	preprocessing stage
S18	Improving the effectiveness of traceability link recovery using hierarchical bayesian networks	Hierarchical Bayesian Network (HBN)	link generation stage
S19	Traceability Link Recovery between Requirements and Models using an Evolutionary Algorithm Guided by a Learning to Rank Algorithm: Train control and management case	FNN RNN RankBoost	link generation stage
S20	Information retrieval versus deep learning approaches for generating traceability links in bilingual projects	Word2vec FastText BERT	preprocessing stage
S21	Issue Link Label Recovery and Prediction for Open Source Software	FastText Logistic Regression (LR) Random Forest (RF) Neural Network (NN)	preprocessing stage link generation stage

		ZeroR	
S22	Large Scale Evaluation of Natural Language Processing Based Test-to-Code Traceability Approaches	Doc2vec	preprocessing stage
S23	Leveraging Historical Associations between Requirements and Source Code to Identify Impacted Classes	Decision Tree (DT) Random Forest (RF) Logistic Regression (LR) Naive Bayes (NB) Bagging	link generation stage
S24	On the effect of incompleteness to check requirement-to-method traces	Decision Tree (DT) Random Forest (RF) Naive Bayes (NB) K nearest neighbors (KNN)	link generation stage
S25	On the relationship between similar requirements and similar software	Doc2vec FastText BERT Universal Sentence Encoder (USE)	preprocessing stage
S26	Tracing with Less Data: Active Learning for Classification-Based Traceability Link Recovery	Active Learning Random Forest (RF)	link generation stage
S27	Semantically Enhanced Software Traceability Using Deep Learning Techniques	Word2vec RNN LSTM Bi-LSTM GRU Bi-GRU	preprocessing stage
S28	Semi-Automated Feature Traceability with Embedded Annotations	SVM K nearest neighbors (KNN) Decision Tree (DT)	link generation stage
S29	Source Code Level Word Embeddings in Aiding Semantic Test-to-Code Traceability	Doc2vec	preprocessing stage
S30	Tackling the term-mismatch problem in automated trace retrieval	Decision Tree (DT) Naive Bayes (NB)	link generation stage
S31	TCTracer: Establishing test-to-code traceability links using dynamic and static techniques	FNN	link generation stage
S32	Toward accurate link between code and software documentation	Word2vec GBDT	preprocessing stage link generation stage
S33	Towards feature-aware retrieval of refinement traces	Graph Clustering	link refinement stage
S34	Traceability Transformed_ Generating more Accurate Links with Pre-Trained BERT Models	BERT LSTM Bi-GRU	preprocessing stage link generation stage
S35	Towards the automatic classification of traceability links	Decision Tree (DT) Random Forest (RF) K nearest neighbors (KNN) Naive Bayes (NB)	link generation stage
S36	Tracing Requirements as a Problem of	SVM	link generation stage

	Machine Learning	Single-link clustering	preprocessing stage
S37	Traceability in the wild: automatically augmenting incomplete trace links	Naive Bayes (NB) Decision Tree (DT) Random Forest (RF)	link generation stage

1.3 Extracted Data for RQ3

Index	Source Artifact (number)	Target Artifact (number)	Datasets (true link number)	Evidence Level
S1	Use Case	Code	eTour	Level 1: Evaluation conducted in academic context (0.6)
S2	High-level requirement Use case Use case Use case	Low-level requirement Code Interaction Diagrams Test Case	CM1-NASA GANNT eTOUR iTrust EasyClinic	Level 1: Evaluation conducted in academic context (0.6)
S3	Requirement	Code	SmartTrip * SafeDrink * BlueWallet *	Level 2: Evaluation conducted in industry context (1.0)
S4	Requirement Requirement	Use case Design	Pine CM1SUB	Level 1: Evaluation conducted in academic context (0.6)
S5	High-level requirements	Design	ARC-IT	Level 2: Evaluation conducted in industry context (1.0)
S6	Bug Report	Test Case	Mozilla Firefox	Level 1: Evaluation conducted in academic context (0.6)
S7	High-level requirement Use Case Test Case Test Case Interaction Diagram	Low-level requirement Code Use Case Code Test Case	eAnci SMOS MODIS EasyClinic eTour	Level 1: Evaluation conducted in academic context (0.6)
S8	High-level requirement Use Case Test Case Test Case Interaction Diagram Interaction Diagram Interaction Diagram	Low-level requirement Code Use Case Code Test Case Code Use Case	eAnci SMOS MODIS EasyClinic eTour iTrust	Level 1: Evaluation conducted in academic context (0.6)
S9	High-level requirement Use Case Test Case Test Case Interaction Diagram Interaction Diagram Interaction Diagra	Low-level requirement Code Use Case Code Test Case Code Use Case	eAnci SMOS EasyClinic eTour iTrust CM-1	Level 1: Evaluation conducted in academic context (0.6)
S10	Requirement	Requirement	Dataset1 * Dataset2 *	Level 1: Evaluation conducted in academic context (0.6)
S11	Use Case	Code	eTour SMOS Albergate	Level 1: Evaluation conducted in academic context (0.6)

			eAnci	
S12	Requirement	Code	SmartTrip * SafeDrink * BlueWallet *	Level 2: Evaluation conducted in industry context (1.0)
S13	Requirement Requirement	Requirement Design	AIRFLOW ANY23 DASHBUILDER DROOLS IMMUTANT JBTM MODIS CM-1	Level 1: Evaluation conducted in academic context (0.6)
S14	Use Case	Code	eAnci eTour SMOS	Level 1: Evaluation conducted in academic context (0.6)
S15	Requirement Use Case Use Case Use Case	Requirement Code Test Case Interaction Diagram	GANNT CM1-NASA eTour iTrust EasyClinic	Level 1: Evaluation conducted in academic context (0.6)
S16	Requirement Use Case Use Case Use Case Use Case Test Case Test Case Interaction Diagram Interaction Diagram Interaction Diagram Interaction Diagram Code Code	Requirement Code Use Case Test Case Interaction Diagram Test Case Code Interaction Diagram Code Test Case Use Case Code Test Case	Selex SI eTour EasyClinic	Level 2: Evaluation conducted in industry context (1.0)
S17	Test Case	Code	Commons Lang Commons Math JfreeChart MONDRIAN	Level 1: Evaluation conducted in academic context (0.6)
S18	Requirement Requirement Use Case	Code Test Case Code	Albergate EBT LibEST eTour SMOS iTrust	Level 2: Evaluation conducted in industry context (1.0)
S19	Requirement	Model	CAF	Level 2: Evaluation conducted in industry context (1.0)
S20	Commit	Issue	Arthas bk-cmdb Canal Druid	Level 1: Evaluation conducted in academic context (0.6)

			Emmagee Nacos NCNN Pegasus QMUI Android QMUI IOS Rax San Weui xLua Konlpy Cica Aws-berline	
S21	Issue	Issue	AMBARI FLEX HIVE	Level 1: Evaluation conducted in academic context (0.6)
S22	Test Case	Code	ArgoUML Commons Lang Commons Math Gson JfreeChart Joda-Time MONDRIAN PMD	Level 1: Evaluation conducted in academic context (0.6)
S23	Requirement	Code	Accumulo Ignite Isis Tika	Level 1: Evaluation conducted in academic context (0.6)
S24	Requirement	Code	Chess Gantt iTrust JHotDraw	Level 2: Evaluation conducted in industry context (1.0)
S25	Requirement Requirement	Requirement Code	A * B *	Level 2: Evaluation conducted in industry context (1.0)
S26	High-level requirement Use Case Test Case Test Case Interaction Diagram Interaction Diagram Interaction Diagram	Low-level requirement Code Use Case Code Test Case Code Use Case	eAnci SMOS MODIS EasyClinic eTour iTrust	Level 1: Evaluation conducted in academic context (0.6)
S27	Requirement	Design	PTC	Level 2: Evaluation conducted in industry context (1.0)
S28	Code	Code	Clafer Tools	Level 1: Evaluation conducted in academic context (0.6)
S29	Test Case	Code	Commons Lang Commons Math JfreeChart MONDRIAN	Level 1: Evaluation conducted in academic context (0.6)

S30	Regulatory code	Requirement	Care2x CCHIT ClearHealth Physician iTrust Trial Implementations PatientOS PracticeOne Lauesen WorldVistA	Level 1: Evaluation conducted in academic context (0.6)
S31	Test Case	Code	Apache Ant Commons IO Commons Lang JfreeChart Gson	Level 1: Evaluation conducted in academic context (0.6)
S32	Code	Software documentation	Lucene	Level 1: Evaluation conducted in academic context (0.6)
S33	Requirement Use Case Feature	Use Case Test Case Use Case	CM-1 EasyClinic Waterloo	Level 2: Evaluation conducted in industry context (1.0)
S34	Commit	Issue	CodeSearchNet Pgcli Flask Keras	Level 1: Evaluation conducted in academic context (0.6)
S35	Requirement Use Case Test Case Interaction Diagram Interaction Diagram Test Case Interaction Diagram	Requiremen Code Code Test Case Use Case Use Case Code	CM-1 eAnci eTour SMOS iTrust EasyClinic	Level 1: Evaluation conducted in academic context (0.6)
S36	Requirement	Use case	Pine	Level 1: Evaluation conducted in academic context (0.6)
S37	Commit	Issue	Maven Derby Infinispan Groovy Pig Drools	Level 1: Evaluation conducted in academic context (0.6)

* present that author uses a pseudonym of the name of dataset for confidentiality agreements

1.4 Extracted Data for RQ4

Index	Title	Measures	Evidence Level
S1	An extended knowledge representation learning approach for context-based traceability link recovery	Precision Recall F-Measure	Level 3: Evidence obtained from academic studies (0.6).
S2	An Improved Approach to Traceability Recovery	Precision	Level 3: Evidence obtained from academic studies (0.6).

	Based on Word Embeddings	Recall F-Measure MAP MRR Running Time	
S3	An information theoretic approach for extracting and tracing non-functional requirements	Precision Recall	Level 3: Evidence obtained from academic studies (0.6).
S4	Application of reinforcement learning to requirements engineering requirements tracing	Precision Recall F-Measure	Level 3: Evidence obtained from academic studies (0.6).
S5	ATLaS: A Framework for Traceability Links Recovery Combining Information Retrieval and Semi-Supervised Techniques	Precision Recall F-Measure	Level 4: Evidence obtained from industrial studies (0.6).
S6	Traceability recovery between bug reports and test cases-a Mozilla Firefox case study	Recall Precision F-Measure REI	Level 4: Evidence obtained from industrial studies (0.6).
S7	Automatic traceability link recovery via active learning	Precision Recall F-Measure	Level 3: Evidence obtained from academic studies (0.6).
S8	Automatic Traceability Maintenance via Machine Learning Classification	Precision Recall F-Measure	Level 3: Evidence obtained from academic studies (0.6).
S9	Automating traceability link recovery through classification	Recall (TPR) FPR	Level 3: Evidence obtained from academic studies (0.6).
S10	Clustering for Traceability Managing in System Specifications	Precision	Level 3: Evidence obtained from academic studies (0.6).
S11	Combining Machine Learning and Logical Reasoning to Improve Requirements Traceability Recovery	Precision Recall F-Measure	Level 3: Evidence obtained from academic studies (0.6).
S12	Detecting, classifying, and tracing non-functional software requirements	Precision Recall	Level 3: Evidence obtained from academic studies (0.6).
S13	Enhancing Automated Requirements Traceability by Resolving Polysemy	Precision Recall F-Measure	Level 3: Evidence obtained from academic studies (0.6).
S14	Enhancing software model encoding for feature location approaches based on machine learning techniques	TP FP	Level 3: Evidence obtained from academic studies (0.6).
S15	Enhancing Unsupervised Requirements Traceability with Sequential Semantics	Precision Recall F-Measure	Level 3: Evidence obtained from academic studies (0.6).
S16	Estimating the number of remaining links in traceability recovery	MRE MAE	Level 3: Evidence obtained from academic studies (0.6).
S17	Evaluation of Textual Similarity Techniques in Code Level Traceability	Precision	Level 3: Evidence obtained from academic studies (0.6).
S18	Improving the effectiveness of traceability link recovery using hierarchical bayesian networks	Precision Recall Average Precision (AP)	Level 4: Evidence obtained from industrial studies (0.6).

S19	Traceability Link Recovery between Requirements and Models using an Evolutionary Algorithm Guided by a Learning to Rank Algorithm: Train control and management case	Recall Precision F-Measure Matthews Correlation Coefficient	Level 4: Evidence obtained from industrial studies (0.6).
S20	Information retrieval versus deep learning approaches for generating traceability links in bilingual projects	Average Precision (AP) F-Measure	Level 3: Evidence obtained from academic studies (0.6).
S21	Issue Link Label Recovery and Prediction for Open Source Software	F-Measure	Level 4: Evidence obtained from industrial studies (0.6).
S22	Large Scale Evaluation of Natural Language Processing Based Test-to-Code Traceability Approaches	Precision	Level 3: Evidence obtained from academic studies (0.6).
S23	Leveraging Historical Associations between Requirements and Source Code to Identify Impacted Classes	Precision Recall F-Measure	Level 3: Evidence obtained from academic studies (0.6).
S24	On the effect of incompleteness to check requirement-to-method traces	Precision Recall F-Measure	Level 4: Evidence obtained from industrial studies (0.6).
S25	On the relationship between similar requirements and similar software	None	Level 3: Evidence obtained from academic studies (0.6).
S26	Tracing with Less Data: Active Learning for Classification-Based Traceability Link Recovery	F-Measure	Level 3: Evidence obtained from academic studies (0.6).
S27	Semantically Enhanced Software Traceability Using Deep Learning Techniques	Precision Recall MAP	Level 3: Evidence obtained from academic studies (0.6).
S28	Semi-Automated Feature Traceability with Embedded Annotations	Precision Recall F-Measure	Level 4: Evidence obtained from industrial studies (0.6).
S29	Source Code Level Word Embeddings in Aiding Semantic Test-to-Code Traceability	Precision	Level 3: Evidence obtained from academic studies (0.6).
S30	Tackling the term-mismatch problem in automated trace retrieval	Precision Recall F-Measure MAP	Level 3: Evidence obtained from academic studies (0.6).
S31	TCTracer: Establishing test-to-code traceability links using dynamic and static techniques	Precision Recall F-Measure MAP AUC	Level 3: Evidence obtained from academic studies (0.6).
S32	Toward accurate link between code and software documentation	Precision Recall F-Measure TNR	Level 3: Evidence obtained from academic studies (0.6).
S33	Towards feature-aware retrieval of refinement traces	Precision Recall Average Precision (AP)	Level 4: Evidence obtained from industrial studies (0.6).
S34	Traceability Transformed: Generating more Accurate	F-Measure	Level 3: Evidence obtained from academic studies (0.6).

	Links with Pre-Trained BERT Models	MAP MRR Precision Running Time	
S35	Towards the automatic classification of traceability links	Recall FPR	Level 3: Evidence obtained from academic studies (0.6).
S36	Tracing Requirements as a Problem of Machine Learning	Recall Precision F-Measure	Level 3: Evidence obtained from academic studies (0.6).
S37	Traceability in the wild: automatically augmenting incomplete trace links	Precision Recall F-Measure	Level 3: Evidence obtained from academic studies (0.6).

1.5 The information of Datasets and the studied papers which used the datasets

Dataset Name	Source Artifacts (Number)	Target Artifacts (Number)	True link	Scale (Total)	Source Link	Freq	Primary researches
eTour	Use Case (58)	Code (116)	336	(174)	http://www.coest.org/	12	[S1] [S2] [S7] [S8] [S9] [S11] [S14] [S15] [S16] [S18] [S26] [S35]
	Use Case (58)	Code (116)	308	(174)			
	Use Case (58)	Code (116)	385	(174)			
	Use Case (58)	Code (116)	366	(174)			
	Use Case (Unclear)	Code (Unclear)	365	Unclear			
EasyClinic	Use Case (30)	Code (47)	93	(77)	http://www.coest.org/	10	[S2] [S7] [S8] [S9] [S15] [S16] [S33] [S35] [S26]
	Use Case (30)	Test Case (63)	63	(93)			
	Use Case (30)	Test Case (47)	63	(77)			
	Use Case (30)	Interaction Diagram (20)	26	(50)			
	Use Case (30)	Use Case (30)	53	(60)			
	Test Case (63)	Test Case (63)	578	(126)			
	Test Case (63)	Code (47)	204	(110)			
	Test Case (Unclear)	Use Case (Unclear)	63	Unclear			
	Interaction Diagram (20)	Use Case (30)	26	(50)			
	Interaction Diagram (20)	Test Case (63)	83	(83)			
	Interaction Diagram (20)	Code (47)	69	(67)			
	Interaction Diagram (20)	Interaction Diagram (20)	59	(40)			
	Code (47)	Code (47)	69	(94)			
	Code (47)	Test Case (63)	202	(110)			
iTrust	Use Case (131)	Code (367)	534	(498)	http://www.coest.org/	10	[S2] [S8] [S9] [S15] [S18] [S24] [S26] [S35] [S30]
	Requirement (131)	Code (367)	399	(498)			
	Requirement (131)	Code (332)	535	(463)			
	Requirement (34)	Code (4913)	307	(4947)			
	Use Case (Unclear)	Code (Unclear)	58	Unclear			
SMOS	Use Case (67)	Code (100)	1045	(167)	http://www.coest.org/	8	[S7] [S8] [S9] [S11] [S14] [S18] [S35]
	Use Case (67)	Code (100)	1044	(167)			

							[S26]
CM-1	High-level requirement (235)	Low-level design document (220)	Unclear	(455)	http://www.coest.org/	8	[S2] [S4] [S9] [S13] [S15] [S33] [S35]
	High-level requirement (22)	Low-level requirement (53)	45	(75)			
	Requirement (22)	Design (46)	46	(68)			
	Requirement (22)	Design (53)	45	(75)			
	Requirement (Unclear)	Use Case (Unclear)	Unclear	Unclear			
eAnci	Use Case (140)	Code (55)	567	(195)	http://www.coest.org/	7	[S7] [S8] [S9] [S11] [S14] [S35] [S26]
	Use Case (Unclear)	Code (Unclear)	554	Unclear			
Commons Lang	Test Case (2473)	Code (596)	Unclear	(3069)	https://github.com/apache/commons-lang	4	[S17] [S22] [S29] [S31]
	Test Case (3061)	Code (3111)	163	(6172)			
JfreeChart	Test Case (2239)	Code (953)	Unclear	(3192)	https://github.com/jfree/jfreechart	4	[S17] [S22] [S29] [S31]
	Test Case (2244)	Code (9053)	432	(11297)			
MODIS	High-level requirement (19)	Low-level requirement (49)	41	(68)	http://promise.site.uottawa.ca/SERepository	4	[S7] [S8] [S13] [S26]
MONDRIAN	Test Case (1546)	Code (1626)	Unclear	(3172)	https://github.com/pentaho/mondrian	3	[S17] [S22] [S29]
Commons Math	Test Case (3493)	Code (2033)	Unclear	(5526)	https://github.com/apache/commons-math	3	[S17] [S22] [S29]
Albergate	Use Case (17)	Code (55)	54	(72)	http://www.coest.org/	2	[S11] [S18]
	Requirement (55)	Code (17)	53	(72)			
Gson	Test Case (924)	Code (757)	Unclear	(1681)	https://github.com/google/gson	2	[S22] [S31]
	Test Case (1006)	Code (635)	55	(1641)			
GANNT	High-level requirement (17)	Low-level requirement (69)	68	(86)	http://www.coest.org/	2	[S15] [S2]
CCHIT	Code (453)	Requirement (958)	534	(1411)	http://www.coest.org/	2	[S30]
	Requirement (Unclear)	Requirement (Unclear)	1046	Unclear			
EBT	Requirement (40)	Test Case (25)	51	(65)	http://www.coest.org/	1	[S18]
	Requirement (40)	Code (50)	98	(90)			
LibEST	Requirement (59)	Code (11)	204	(70)	http://sarec.nd.edu/coest/datasets.html	1	[S18]
	Requirement (59)	Test Case (18)	352	(77)			
Selex SI	Requirement (Unclear)	Requirement (Unclear)	138	(2500)	http://www.finmeccanica.com/en/home	1	[S16]
AMBARI	Issue (Unclear)	Issue (Unclear)	942	(1512)	http://ambari.apache.org	1	[S21]
FLEX	Issue (Unclear)	Issue (Unclear)	247	(362)	http://flex.apache.org	1	[S21]
HIVE	Issue (Unclear)	Issue (Unclear)	5811	(6730)	http://hive.apache.org	1	[S21]
Chess	Requirement (8)	Code (752)	563	(760)	https://github.com/warpwe/java-chess	1	[S24]
Gantt	Requirement (18)	Code (5013)	343	(5031)	https://sourceforge.net/projects/ganttproject	1	[S24]
JHotDraw	Requirement (21)	Code (6520)	439	(6541)	https://sourceforge.net/projects/jhotdraw	1	[S24]
CodeSearch Net	Commit (Unclear)	Issue (Unclear)	Unclear	Unclear	https://github.com/github/CodeSearchNet	1	[S34]
Pgcli	Commit (531)	Issue (522)	530	(1053)	https://zenodo.org/record/4511291#.YB3tjyj0mbg	1	[S34]
Flask	Commit (752)	Issue (739)	753	(1491)		1	[S34]
Keras	Commit (551)	Issue (550)	51	(1101)		1	[S34]

ARC-IT	Requirement (2395)	System Functions (802)	2395	(3197)	https://local.iteris.com/arc-it/index.html	1	[S5]
Commons IO	Test Case (994)	Code (1246)	97	(2240)	https://commons.apache.org/proper/commons-io/	1	[S31]
Apache Ant	Test Case (1830)	Code (10477)	79	(12307)	https://ant.apache.org/	1	[S31]
Mozilla Firefox	Bug Report (34)	Test Case (113)	514	(147)	https:// github.com/ guilhermemg/trace-links-tc-br	1	[S6]
Arthas	Commit (122)	Issue (167)	167	(289)	https://doi.org/10.5281/zenodo.3713256	1	[S20]
bk-cmdb	Commit (895)	Issue (1178)	1179	(2073)		1	[S20]
Canal	Commit (232)	Issue (273)	273	(505)		1	[S20]
Druid	Commit (1092)	Issue (1161)	1161	(2253)		1	[S20]
Emmagee	Commit (31)	Issue (32)	32	(63)		1	[S20]
Nacos	Commit (132)	Issue (161)	161	(293)		1	[S20]
NCNN	Commit (97)	Issue (99)	99	(196)		1	[S20]
Pegasus	Commit (160)	Issue (160)	160	(320)		1	[S20]
QMUI Android	Commit (70)	Issue (71)	71	(141)		1	[S20]
QMUI IOS	Commit (32)	Issue (35)	35	(67)		1	[S20]
Rax	Commit (560)	Issue (571)	571	(1131)		1	[S20]
San	Commit (186)	Issue (275)	275	(461)		1	[S20]
Weui	Commit (154)	Issue (159)	159	(313)		1	[S20]
xLua	Commit (52)	Issue (52)	52	(104)		1	[S20]
Konlpy	Commit (32)	Issue (33)	33	(65)		1	[S20]
Cica	Commit (25)	Issue (27)	27	(52)		1	[S20]
Aws-berline	Commit (74)	Issue (74)	74	(148)		1	[S20]
DASHBUILDER	Requirement (Unclear)	Requirement (Unclear)	Unclear	(85)	https://issues.jboss.org/browse/DASHBUILDE	1	[S13]
Maven	Commit (8205)	Issue (4728)	Unclear	(12933)	https://issues.apache.org/jira/browse/MNG	1	[S37]
Derby	Commit (4468)	Issue (3608)	Unclear	(8076)	https://issues.apache.org/jira/browse/DERBY	1	[S37]
Groovy	Commit (1754)	Issue (2709)	Unclear	(4463)	https://issues.apache.org/jira/browse/GROOVY	1	[S37]
JBTM	Requirement (Unclear)	Requirement (Unclear)	Unclear	(1575)	https://issues. jboss.org/browse/JBTM	1	[S13]
Accumulo	Requirement (145)	Code (593)	3412	(738)	http://isis.apache.org	1	[S23]
Ignite	Requirement (41)	Code (668)	15569	(709)	https://ignite.apache.org/	1	[S23]
Isis	Requirement (252)	Code (2424)	11850	(2676)	http://isis.apache.org	1	[S23]
Tika	Requirement (49)	Code (72)	248	(121)	http://tika.apache.org	1	[S23]
Care2x	Requirement (Unclear)	Requirement (Unclear)	44	Unclear	http://www.care2x.org	1	[S30]
ClearHealth	Requirement (Unclear)	Requirement (Unclear)	44	Unclear	e http://www.clear-health.com	1	[S30]
Physician	Requirement (Unclear)	Requirement (Unclear)	147	Unclear	hmss.org/content/files/CTC_use_Case.pdf	1	[S30]
Trial Implementations	Requirement (Unclear)	Requirement (Unclear)	100	Unclear	http://healthit.hhs.gov	1	[S30]

PatientOS	Requirement (Unclear)	Requirement (Unclear)	90	Unclear	http://www.patientos.org	1	[S30]
PracticeOne	Requirement (Unclear)	Requirement (Unclear)	34	Unclear	http://www.practiceone.com	1	[S30]
WorldVistA	Requirement (Unclear)	Requirement (Unclear)	66	Unclear	http:/worldvista.org	1	[S30]
Pine	Requirement (49)	Use case (51)	250	(100)		2	[S4] [S36]
	Requirement (49)	Use case (51)	246	(100)			
SafeDrink *	Functional requirement (170)	Code (173)	Unclear	(343)		2	[S3] [S12]
SmartTrip *	Functional requirement (214)	Code (266)	Unclear	(480)		2	[S3] [S12]
BlueWallet *	Functional requirements (184)	Code (374)	Unclear	(558)		2	[S3] [S12]
Drools	Requirement (Unclear)	Requirement (Unclear)	Unclear	(486)		2	[S13] [S37]
	Commit (3735)	Issue (3992)	Unclear	(7727)			
Lauesen	Requirement (Unclear)	Requirement (Unclear)	116	Unclear		1	[S30]
Joda-Time	Test Case (3779)	Code (522)	Unclear	(4301)		1	[S22]
PTC	Requirement (1651)	Design (466)	1387	(2117)		1	[S27]
Lucene	Code (5097)	Software documentation (1899)	2137	(6996)		1	[S32]
A *	Requirement (112)	Requirement (142)	Unclear	(254)		1	[S25]
B *							
ArgoUML	Test Case (554)	Code (2404)	Unclear	(2958)		1	[S22]
Waterloo	Feature (Unclear)	Use Case (Unclear)	Unclear	Unclear		1	[S33]
PMD	Test Case (825)	Code (1608)	Unclear	(2433)		1	[S22]
Clafer Tools	Feature annotation (14000)	Code (Unclear)	Unclear	Unclear		1	[S28]
AIRFLOW	Requirement (Unclear)	Requirement (Unclear)	Unclear	(629)		1	[S13]
ANY23	Requirement (Unclear)	Requirement (Unclear)	Unclear	(182)		1	[S13]
Pig	Commit (4839)	Issue (2012)	Unclear	(6851)		1	[S37]
Infinispan	Commit (4778)	Issue (2058)	Unclear	(6836)		1	[S37]
IMMUTANT	Requirement (Unclear)	Requirement (Unclear)	Unclear	(404)		1	[S13]
CAF	Requirement (Unclear)	Model (Unclear)	Unclear	Unclear		1	[S19]
Dataset1 *	Requirement (762)	Requirement (521)	367	(1283)		1	[S10]
Dataset2 *	Requirement (2060)	Requirement (4188)	817	(6248)		1	[S10]

* present that author uses a pseudonym of the name of dataset for confidentiality agreements

2. Search process record

Database	Number of searches	Number of repetitions in each database	Number of each database (After deleting repetitions)	Number of repetitions in all databases	Total number (After deleting repetitions)
ACM	96	1	69	227	56
Springer	210	0	171		132
Science Direct	136	20	113		80
EI	674	38	596		457

IEEE	324	67	243		240
Total	**1440**	**126**	**1192**		**965**

Excute inclusion/exclusion criteria

Database	apply criteria (ISC1-ISC3, ESC1-ESC4)	apply criteria(ISC4-ISC5, ESC5-ESC6) in title, abstract, keywords	apply criteria(ISC4-ISC5, ESC5-ESC6) in full article	Snowballing	final
ACM	**625**	**184**	**35**	**3**	**38**
Springer					
Science Direct					
EI					
IEEE					
Total					

1.6 Search records

Digital Libraries：

Database	Website
ACM	https://dl.acm.org/
Springer	https://www.springer.com/
Science Direct	https://www.sciencedirect.com/
EI	https://www.engineeringvillage.com/
IEEE	https://ieeexplore.ieee.org/

Search terms：

P1	software traceability	**I1**	machine learning
P2	software trace	**I2**	ML
P3	software tracing	**I3**	supervised learning
P4	traceability link recovery	**I4**	unsupervised learning
		I5	semi-supervised learning
		I6	reinforcement learning

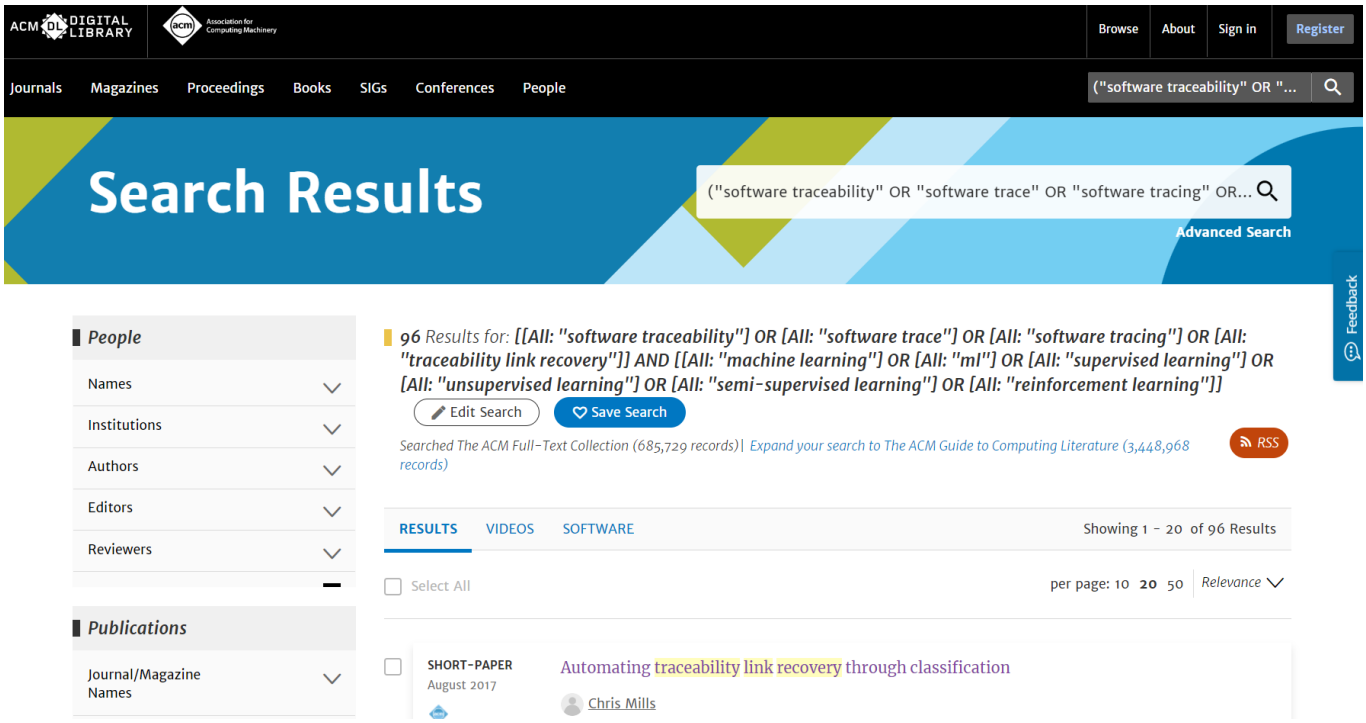
(1) ACM

	Anywhere
(P1 OR P2 OR P3 OR P4) AND (I1 OR I2 OR I3 OR I4 OR I5 OR I6)	96

Advanced search:

("software traceability" OR "software trace" OR "software tracing" OR "traceability link recovery") AND ("machine learning" OR "ML" OR "supervised learning" OR "unsupervised learning" OR "semi-supervised learning" OR "reinforcement learning")

Screenshot of search process in ACM:



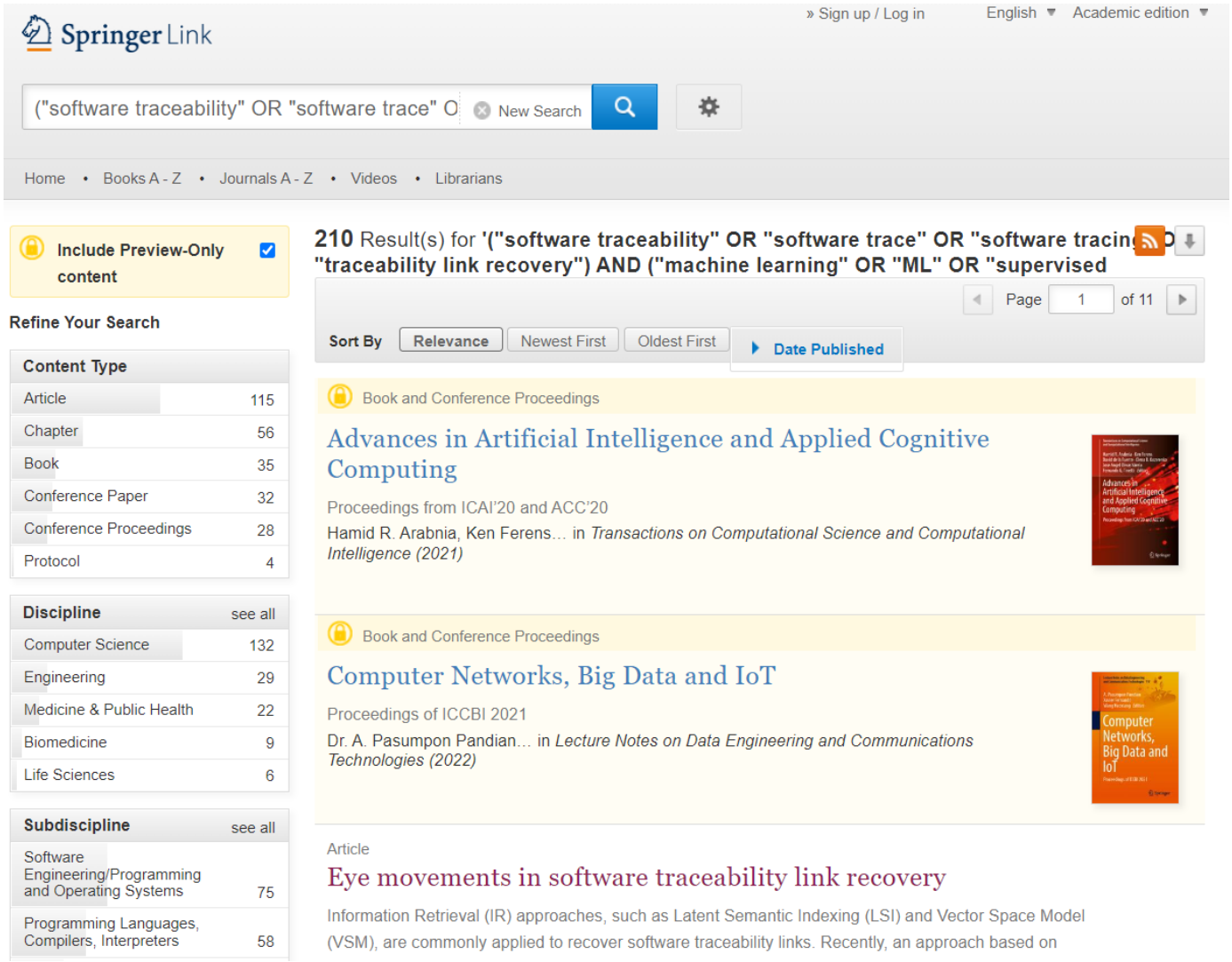
(2) Spinger

	Keywords+Title+Abstract
(P1 OR P2 OR P3 OR P4) AND (I1 OR I2 OR I3 OR I4 OR I5 OR I6)	210

Advanced search:

("software traceability" OR "software trace" OR "software tracing" OR "traceability link recovery") AND ("machine learning" OR "ML" OR "supervised learning" OR "unsupervised learning" OR "semi-supervised learning" OR "reinforcement learning")

Screenshot of search process in Springer:



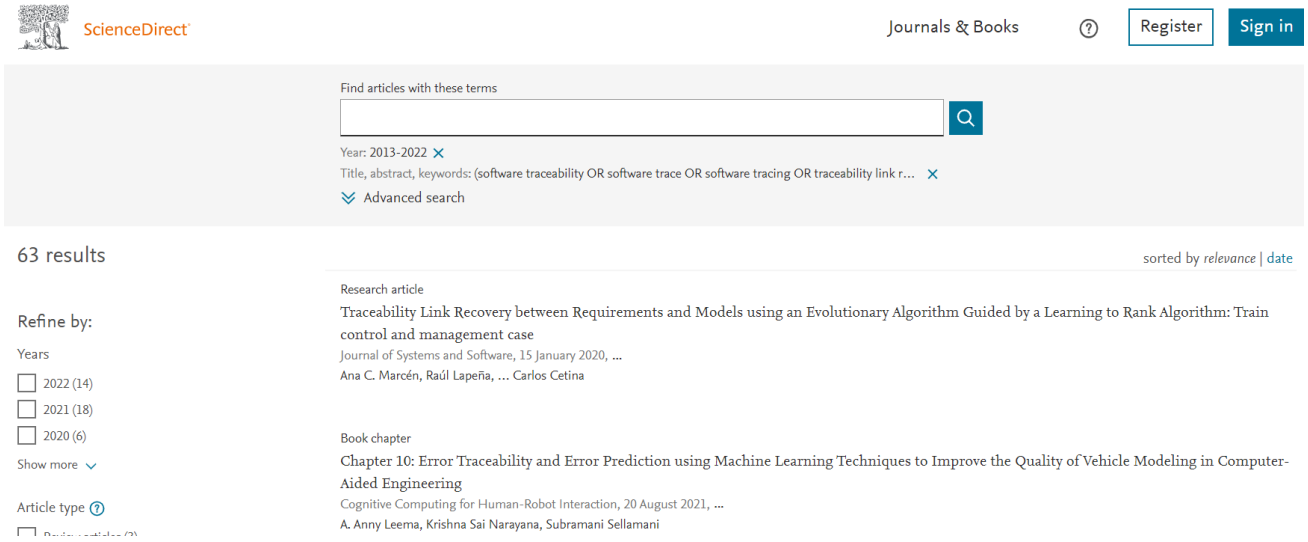
(3) Science Direct

	Title+Abstract+Keywords
(P1 OR P2 OR P3 OR P4) AND (I1)	63
(P1 OR P2 OR P3 OR P4) AND (I2)	59
(P1 OR P2 OR P3 OR P4) AND (I3)	7
(P1 OR P2 OR P3 OR P4) AND (I4)	2
(P1 OR P2 OR P3 OR P4) AND (I5)	1
(P1 OR P2 OR P3 OR P4) AND (I6)	4
Total	136

■ **Advanced search((P1 OR P2 OR P3 OR P4) AND (I1)):**

(software traceability OR software trace OR software tracing OR traceability link recovery) AND (machine learning)

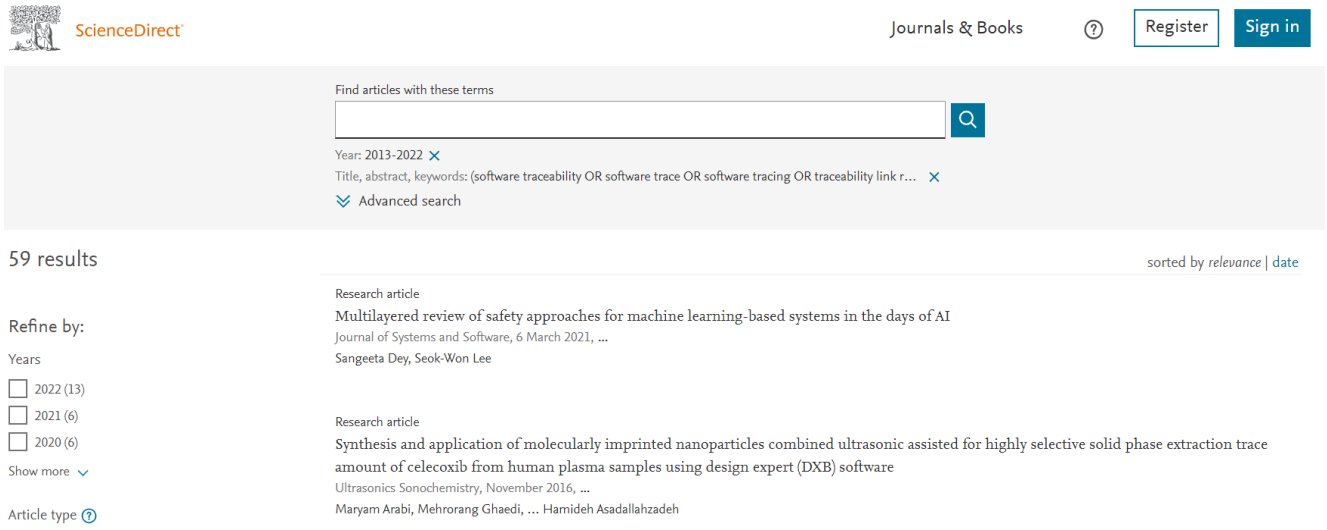
Screenshot of search process in Scienct Direct:



■ **Advanced search((P1 OR P2 OR P3 OR P4) AND (I2)):**

(software traceability OR software trace OR software tracing OR traceability link recovery) AND (ML)

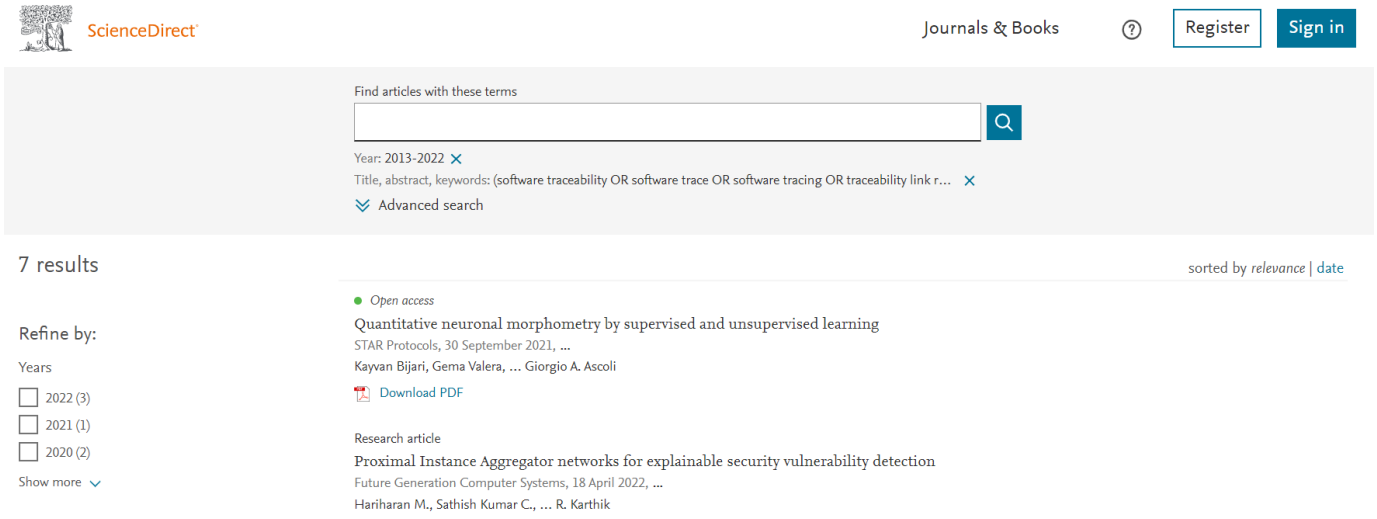
Screenshot of search process in Scienct Direct:



■ **Advanced search((P1 OR P2 OR P3 OR P4) AND (I3)):**

(software traceability OR software trace OR software tracing OR traceability link recovery) AND (supervised learning)

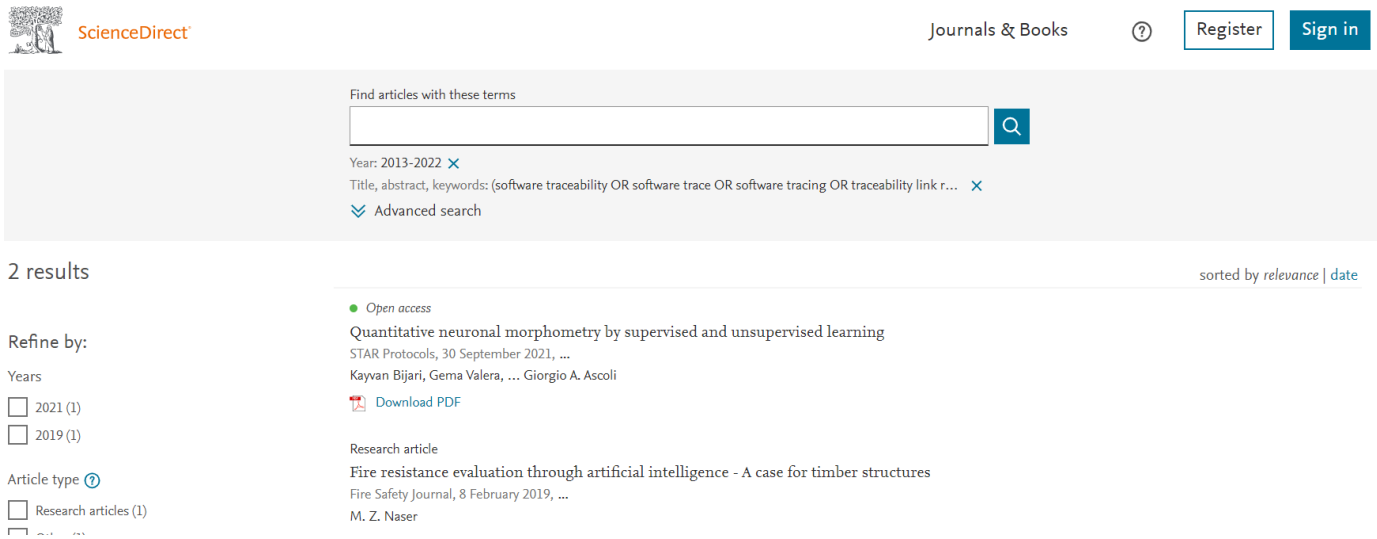
Screenshot of search process in Scienct Direct:



■ **Advanced search((P1 OR P2 OR P3 OR P4) AND (I4)):**

(software traceability OR software trace OR software tracing OR traceability link recovery) AND (unsupervised learning)

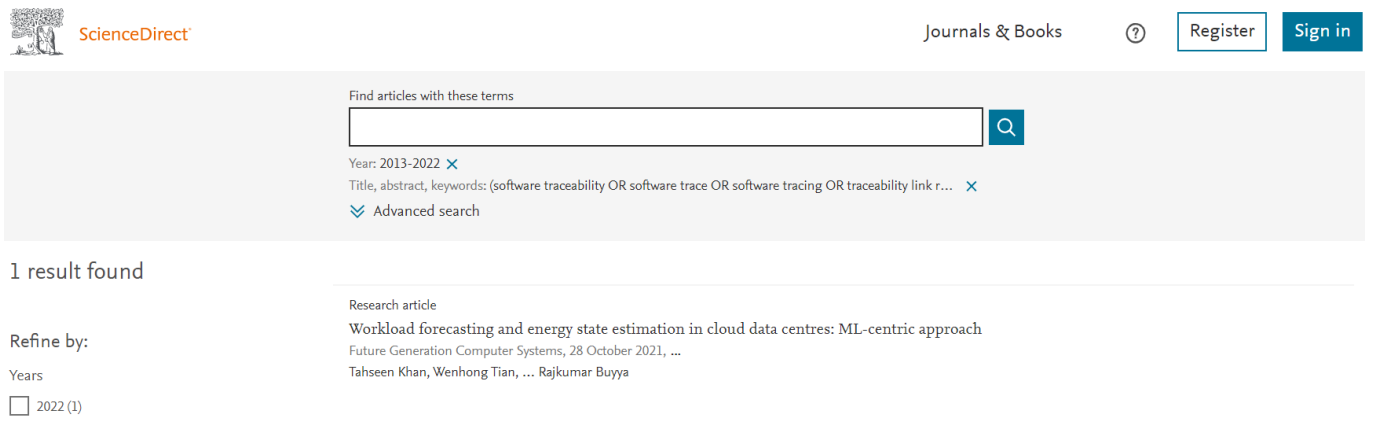
Screenshot of search process in Scienct Direct:



■ **Advanced search((P1 OR P2 OR P3 OR P4) AND (I5)):**

(software traceability OR software trace OR software tracing OR traceability link recovery) AND (semi-supervised learning)

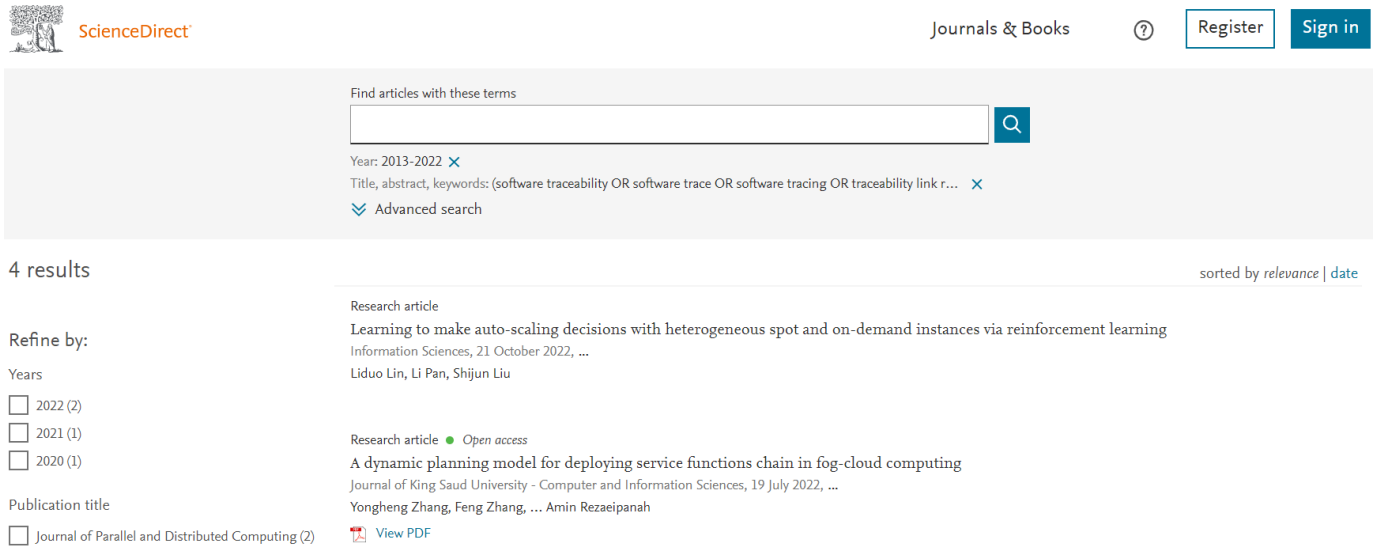
Screenshot of search process in Scienct Direct:



■ **Advanced search((P1 OR P2 OR P3 OR P4) AND (I6)):**

(software traceability OR software trace OR software tracing OR traceability link recovery) AND (reinforcement learning)

Screenshot of search process in Scienct Direct:



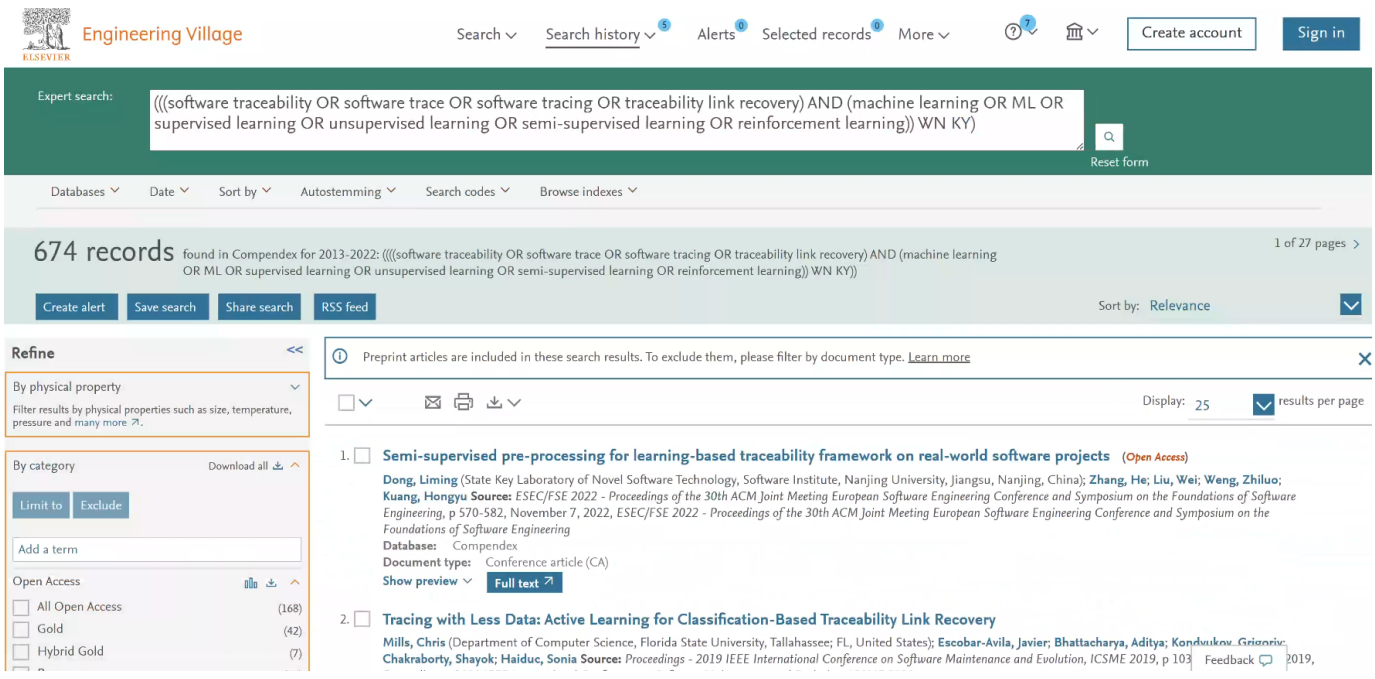
(4) EI

	Subject/Title/Abstract
(P1 OR P2 OR P3 OR P4) AND (I1 OR I2 OR I3 OR I4 OR I5 OR I6)	674

Expert search:

(software traceability OR software trace OR software tracing OR traceability link recovery) AND (machine learning OR ML OR supervised learning OR unsupervised learning OR semi-supervised learning OR reinforcement learning)

Screenshot of search process in EI:



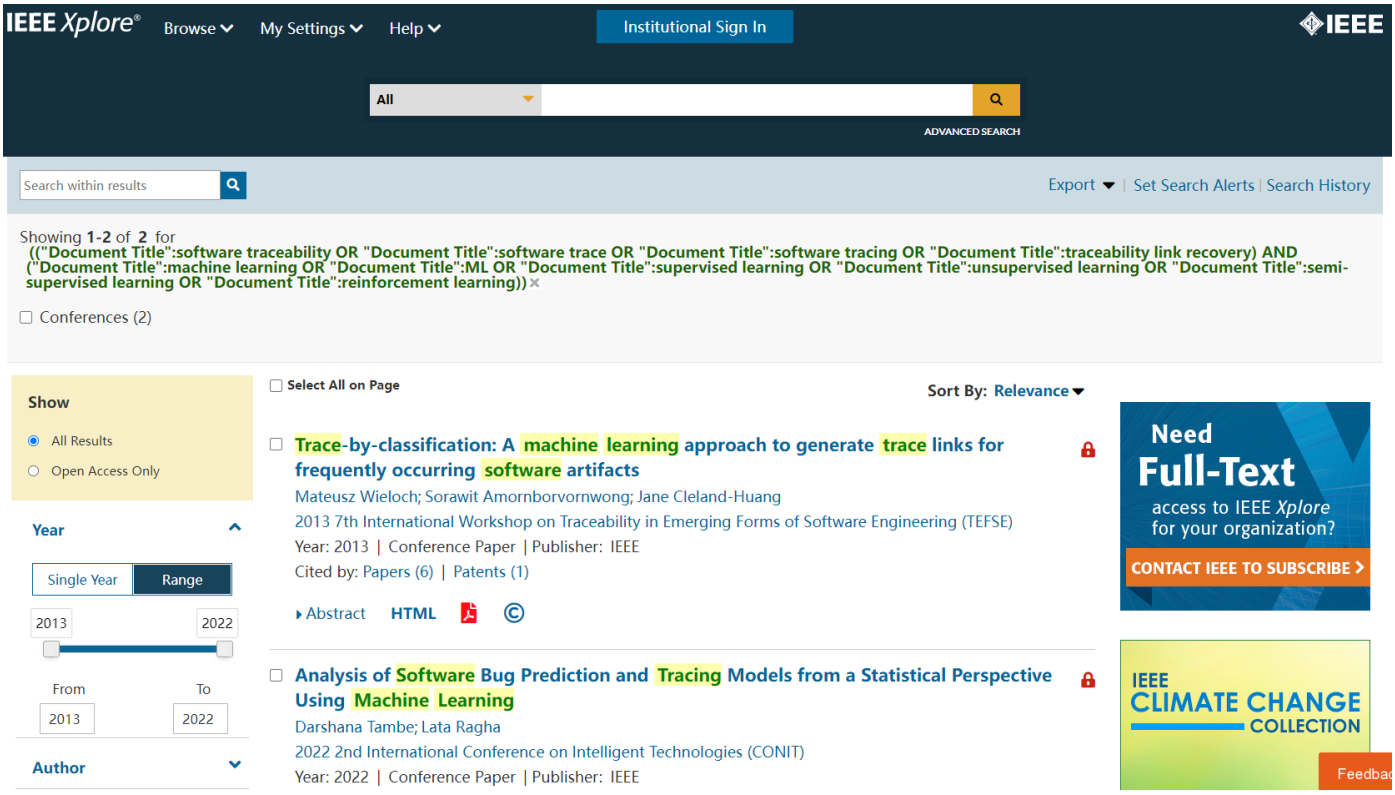
(5) IEEE

	Title	Abstract	Index terms
(P1 OR P2 OR P3 OR P4) AND (I1 OR I2 OR I3 OR I4 OR I5 OR I6)	2	159	163
Total	324		

■ **Command Search(Title):**

("Document Title":software traceability OR "Document Title":software trace OR "Document Title":software tracing OR "Document Title":traceability link recovery) AND ("Document Title":machine learning OR "Document Title":ML OR "Document Title":supervised learning OR "Document Title":unsupervised learning OR "Document Title":semi-supervised learning OR "Document Title":reinforcement learning)

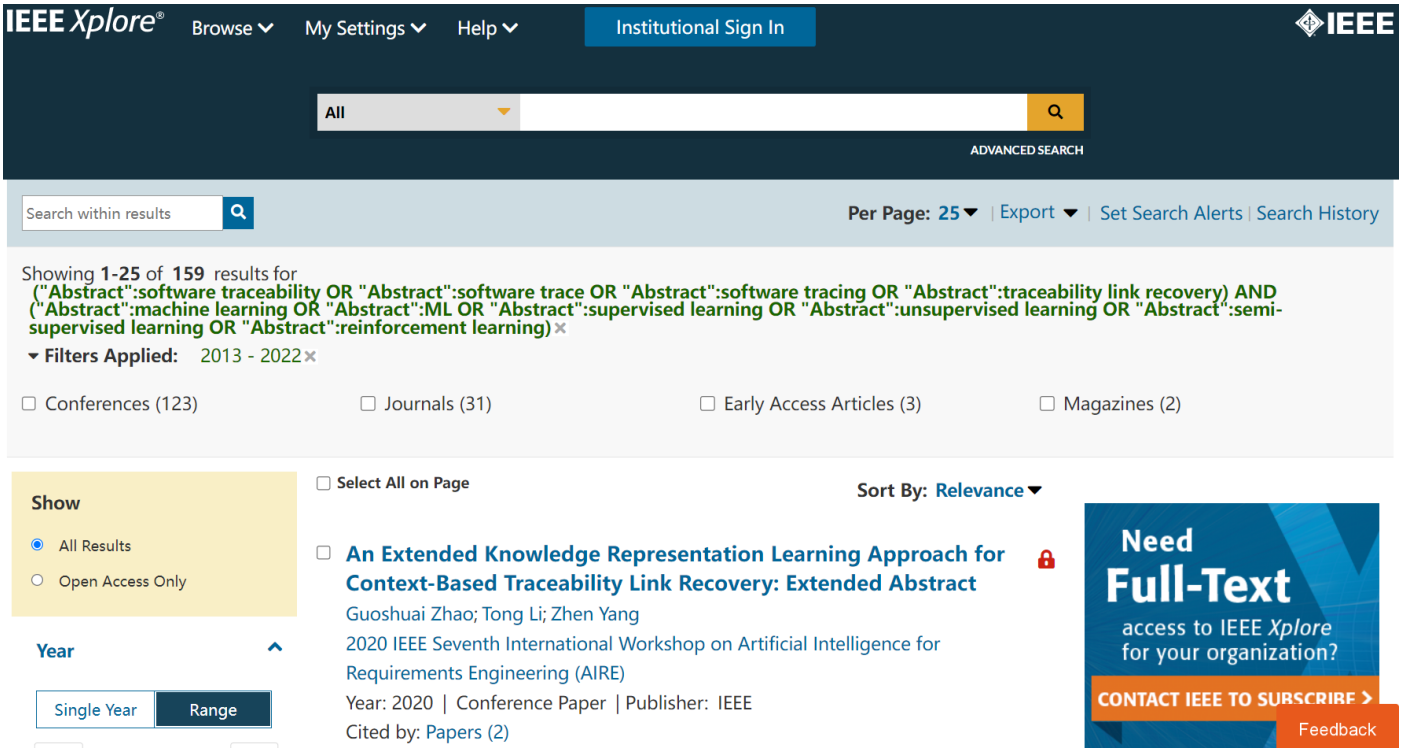
Screenshot of search process in IEEE:



■ **Command Search(Abstract):**

("Abstract":software traceability OR "Abstract":software trace OR "Abstract":software tracing OR "Abstract":traceability link recovery) AND ("Abstract":machine learning OR "Abstract":ML OR "Abstract":supervised learning OR "Abstract":unsupervised learning OR "Abstract":semi-supervised learning OR "Abstract":reinforcement learning)

Screenshot of search process in IEEE:



■ **Command Search(Index Terms):**

("Index Terms":software traceability OR "Index Terms":software trace OR "Index Terms":software tracing OR "Index Terms":traceability link recovery) AND ("Index Terms":machine learning OR "Index Terms":ML OR "Index Terms":supervised learning OR "Index Terms":unsupervised learning OR "Index Terms":semi-supervised learning OR "Index Terms":reinforcement learning)

Screenshot of search process in IEEE: