iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** XI **Month of publication:** November 2021

DOI: https://doi.org/10.22214/ijraset.2021.39076

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue XI Nov 2021- Available at www.ijraset.com

Analysing Predictive Coding Algorithms for Document Review

Aditi Wikhe[1], Jagruti Agrawal[2], Manali Bankar[3], Prerana Baviskar[4]
[1, 2, 3, 4]*Department of Computer Engineering, Pune Institute of Computer Technology, Pune, India*

Abstract: ***Lawsuits and regulatory investigations in today's legal environment demand corporations to engage in increasingly intense data-focused engagements to find, acquire, and evaluate vast amounts of data. In recent years, technology-assisted review (TAR) has become a more crucial part of the document review process in legal discovery. Attorneys now have been using machine learning techniques like text classification to identify responsive information. In the legal domain, text classification is referred to as predictive coding or technology assisted review (TAR). Predictive coding is used to increase the number of relevant documents identified, while reducing human labelling efforts and manual review of documents. Deep learning models mixed with word embeddings have demonstrated to be more effective in predictive coding in recent years. Deep learning models, on the other hand, have a lot of variables, making it difficult and time-consuming for legal professionals to choose the right settings. In this paper, we will look at a few predictive coding algorithms and discuss which one is the most efficient among them.***
Keywords: Technology-assisted-review, predictive coding, machine learning, text classification, deep learning, CNN , Unscented Kalman Filter, Logistic Regression, SVM

I. INTRODUCTION

We came across a case study in the USA, 'Da Silva Moore et al. v. Publicis Groupe Case, 2012' which was amongst the early instances of using predictive coding in the court of law. Gender discrimination lawsuits were filed against an advertising giant and its US subsidiary in this case. The fact that this case was heard by Magistrate Andrew J Peck, a judge who is well-versed in predictive coding technology, was frosting on the cake for proponents of the technique. The case involved more than 3 million documents. The use of predictive coding in the e-discovery process was deemed proper by the Magistrate. He made his ruling based on five factors: (1) the parties' agreement; (2) the number of documents involved; (3) the conclusion that predictive coding is more effective than other options; (4) cost effectiveness and proportionality; and (5) defendants' transparency in the discovery process. As a result, this case from the United States demonstrates that the use of predictive coding in discovery conforms with the numerous laws of civil procedure's discovery obligations. In the legal domain, in case of electronic documents, manual document review is very costly, legal reviewers take approximately $400-$2000 per hour to review the documents. It is also very time consuming and can take about a couple of months or even years for the completion of document review. Attorneys have been utilising text categorization and other machine learning approaches to find pertinent information for years. Predictive coding or technology assisted review are terms used in the legal world to describe text classification (TAR). Predictive coding, adapted from text categorization for litigation support, is an evolving process with identification of responsive documents and changing labeling decisions.

We will be analysing and studying various predictive coding algorithms such as :

1) CNN
2) Unscented Kalman Filter
3) Batch mode learning (Diversity Sampler (DS) and Biased Probabilistic Sampler(BPS))
4) Active learning (Continuous Active Learning, Simple Active Learning, Simple passive Learning)
5) Explainable predictive coding using Logistic Regression

II. LITERATURE REVIEW

Predictive coding typically "begins with no prior knowledge of the dataset and continues until the majority of relevant documents have been found and examined." According to [1],empirical evaluations from their tests showed that Logistic Regression with Unscented Kalman Filter can update the model at a quicker pace and with greater accuracy while reducing labelling cost in the presence of concept drift when compared to reconstructing a model regularly. The authors in [2] proposed two novel methods named Diversity Sampler (DS) and Biased Probabilistic Sampler (BPS). The superiority of their solution over existing methods (Brinker and SVMactive) for the experiment is supported by findings on a series of large-scale real-life legal document collections. By giving responsive snippets justifying the use of predictive coding, the authors believe explainable AI has the potential to considerably enhance the application of text categorization in legal document review problems[3].

©IJRASET: All Rights are Reserved

In accordance with [4], The findings suggest that Simple Passive Learning (SPL) is the least effective TAR approach, casting doubt on not just its effectiveness but also on popularly held TAR assumptions. The findings also reveal that, while Simple Active Learning (SAL) is significantly more effective than SPL, it is typically less effective than Continuous Active Learning (CAL), and only as effective as CAL in a best-case situation unlikely to occur in practice. The impact of various CNN settings on predictive model performance was researched in [5], which looked at the use of CNNs for text categorization in legal cases. It also demonstrates how varied parameter settings affect model performance.

III. METHODOLOGY

A. Batch Mode Learning (DS and BPS).

Both DS and BPS promote diversity, but they differ in how they choose an instance: DS uses deterministic selection, whereas BPS uses probabilistic selection. For the DS technique, they first sort all accessible documents in non-descending order of their distance from the current hyperplane hc, then filter any documents that are identical to the last instance picked in the current batch (we do not choose them in the current batch).In BPS, they create a probability vector and use it to choose a document based on its distance from the current hyperplane.

1) Algorithm for BPS and DS:

Input-
H_c, current hyperplane;
D, available instances;
k, batch size;
t, similarity threshold.
Output-
A batch of k documents to be included in training

```
If Strategy is DS then
   Bc <- EmptySet()
    I <- ArgSort (Distance(hc, D), order=increase)
    while Size (Bc) <k do
       Insert (Bc, I[1])
       S<- GetSimilar(I[1], I, D, t, similarity= cosine)
       I<- Remove(I, S)
else if  Strategy is BPS then
   w<- 1.0/(Distance(hc,D))^2
   w<- Normalize (w)
    I<- List(D)
    while Size (Bc) <k do
       c<-Choose (I, prob=w, num=1)
       Insert(Bc,c)
       S<-GetSimilar(c, I, D, t, similarity=cosine)
       I<-Remove(I,S)
      w<-Normalize(w[I])
return Bc
```

The computational complexity of DS is O(|D| log |D|)
The computational complexity of BPS is O(k · |I| · log |I|)

2) Dataset Used: They used 7 different matters for the experiment. ACQ and MONEY-FX come from the Reuters Dataset1 that is freely available. There are a total of 21, 578 documents in this collection. The records in Matters D1-D4 were assessed for response by a review committee in two separate product liability claims. After filtering out files with no extractable content, D1-D4 has 788, 875 documents, with 50 attorneys working on it. Matter C is from a different dataset that was evaluated by 30 lawyers for a specific litigation. There are 366,999 documents in all.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue XI Nov 2021- Available at www.ijraset.com

B. Explainable Predictive coding using Logistic Regression.

The data set was compiled using 688,294 documents hand categorised as responsive or nonresponsive by attorneys including emails, Microsoft Office documents, PDFs, and other text-based documents from a real-life legal case that is now closed. Only 41,739 of the 688,294 papers are responsive.

The first set of experiments looked at how well the document and reasoning models did at discriminating annotated responsive rationales from not responsive snippets chosen at random from the not responsive documents. Both document and reasoning models were examined in these studies using a test set that included annotated rationales and randomly selected non responsive snippets. As performance indicators, precision and recall were employed.

In the second set of experiments, both document and reasoning models are applied to responsive labelled documents in order to uncover rationales that "explain" the models' responsive conclusion. A responsive document is divided into overlapping snippets in these trials. The performance metric was recall (the proportion of identified rationales). If an annotated justification is contained in the text fragment with the highest score, it is appropriately detected.

C. Effects of CNN Parameters for text Categorization.

Dataset 1 (D1) broadly seeks documents concerning general purpose trading system.

Dataset 2 (D2) broadly seeks documents concerning the current or prospective legality or illegality of the trading of financial products.

Dataset 3 (D3) seeks documents relating to the environmental impact of the company activities.

Table I
Dataset Statistics

Datasets	Total documents	Responsive documents	Non-responsive documents
D1	2500	1040	1460
D2	2102	238	1864
D3	2199	245	1954

1) Experimental Setup: The experiments' goal is to see how changing CNN parameters affects model performance in assisting legal review. In order to compare, we undertake the following experiments: (1) fine-tuned pre-trained word embeddings, no pre trained word embeddings, and static pre-trained word embeddings; (2) different kernel sizes and multiple kernels with different sizes; (3) various combinations of number of filters and features extracted each filter while keeping the total number of features fixed; and (4) different number of filters with 1-max Pooling.

2) Activation function:

a) Rectified Linear Units (ReLU) as the activation function at convolution layer

b) Sigmoid as activation function at the final layer to make binary prediction.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue XI Nov 2021- Available at www.ijraset.com

3) Hyperparameter: A grid search with the optimal predictive performance was used to select the best.

Table II
Parameter Settings

Parameters	Settings
Vocabulary Size	1,000
Sequence Length	4000
Embedding Dimension	50
Pooling Layer	Max Pooling
Number of Filters	varied
Filter Kernel Size	varied
Dropout	0.3

D. Active Learning (CAL, SAL, SPL).

The first 30,000 bytes of each document's ASCII text representation were divided into overlapping 4-byte pieces. Hashing reduced the number of different potential segments from 232 = 4, 294, 967, 296 to 1, 000, 081. Each feature was represented by a binary value: "1" if the feature was present in the document's first 30,000 bytes, and "0" if it was not. They utilised the Sofia-ML implementation of Pegasos SVM5 for the learning algorithm, using the following parameters: "—iterations 2000000 —dimensionality 1100000." They used a batch size of 1000 documents for all the protocols.

In the primary CAL implementation, 1,000 articles were randomly picked from the results of a search using the seed query as the initial training set.

The training-set documents were coded according to the training standard in each iteration, and then used to train Sofia-ML, which was subsequently used to score the remaining documents in the collection. The 1,000 documents with the highest scores were added to the training set, and the procedure was repeated 100 times.

In the primary SAL implementation, the 1,000-document keyword-selected seed set used was identical to that used in CAL. The training-set documents were coded according to the training standard in each iteration, then used to train Sofia-ML, and therefore to score the remaining documents in the collection, much like CAL. Unlike CAL, the 1,000 documents with the lowest magnitude scores were coded and added to the training set, followed by a 100-fold repetition of the procedure.

Throughout the initial SPL implementation, Random selection was employed as some SPL proponents urged. The first training set (which we regard to as the "seed set," despite the fact that many SPL proponents refer to the final training set as the "seed set") comprised of 1,000 randomly picked documents, with 1,000 more randomly selected documents added with each iteration.

©IJRASET: All Rights are Reserved

1) *Dataset used:* Topics 201, 202, 203, and 207 of the TREC 2009 Legal Track Interactive Task - the same Topics that were used to evaluate Cormack and Mojdeh's CAL efforts at TREC – were extracted into four review tasks, labelled Matters 201, 202, 203, and 207. Matters A, B, C, and D are four additional review duties developed from real reviews undertaken during court procedures.

Table III
Dataset Statistics

Matter	Collection Size	No. of relevant documents	Prevalence (%)
201	723537	2454	0;34
202	723537	9514	1.31
203	723537	1826	0.25
207	723537	8850	1.22
A	1118116	4001	0.36
B	409277	6236	1.52
C	293549	1170	0.48
D	405796	15926	3.92

E. Unscented Kalman Filtering.

The collection and processing of documents is the first step. It includes textual content indexing, deduplication, keyword culling, and extraction. The next step is to create an initial model using a sample set of documents. It is chosen from the corpus via keyword search, heuristics, or a combination of both. The documents in the seed set are then rated as responsive or unresponsive by human evaluators. Various NLP techniques, such as tokenization, stop word removal, stemming, and hashing, are used to extract features from text. A preliminary text categorization model is created using logistic regression from the seed set that has been rated. In the third step, the Unscented kalman filter is used to implement active learning in order to continuously improve the model's performance and capture concept drift so that it reflects the reviewers' most recent decisions. Every round of the model update, the filter is used to update the weights of logistic regression. A random sample of n unlabeled documents is taken for each round. This sequence of documents is examined by reviewers. A one-step-ahead prediction is made for each document, and if the prediction uncertainty exceeds the cut-off, a manual human review is requested. With each document reviewed, the model is updated to reflect any changes in pattern, and the updated model is used to predict the next document in the sequence. This process is repeated until the sample's final document is used. Validation is the fourth step. In this step, a validation set is created and labelled based on the reviewer's best current knowledge at the conclusion of the training. On this validation set, the data prevalence and model performance are evaluated and generalised. The final step is production, which entails identifying relevant documents. This final model is used for the rest of the documents and ranked from most likely responsive to least likely responsive.

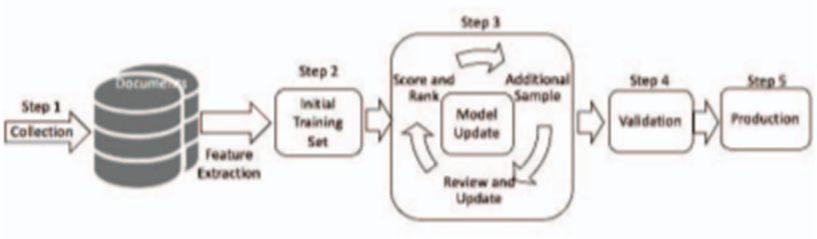


Figure 1: Workflow

©IJRASET: All Rights are Reserved

1) *Dataset Used*: Experiments were carried out on two synthetic streaming text datasets derived from a collection of 20 Newsgroups. Each document has a topic and subtopic that are labelled.

Table IV
Number Of Document By Topics

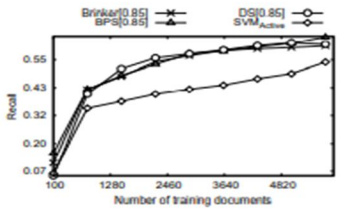
Subtopic	Size
Recreational: Baseball	994
Recreational: Hockey	999
Recreational: Others (Autos, Motorcycles)	1986
Total	3979

TABLE V
Number Of Documents In Each Subtopic

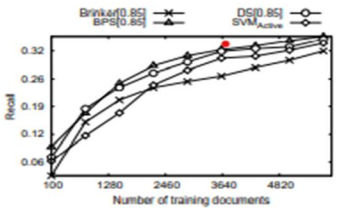
Timeframe	t=0	t=1	t=2	t=3	t=4	t=5	t=6[a]
Number of Hockey Documents	48	97	80	96	104	98	476
Number of Baseball Documents	60	90	102	102	95	75	470
Number of Other Documents	92	190	195	179	178	204	948
Number of Documents	200	377	377	377	377	377	1894

IV. RESULT

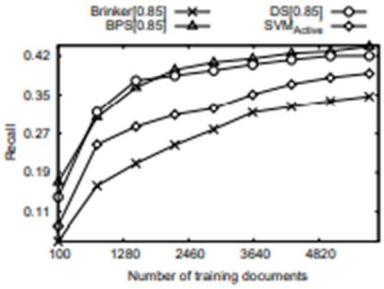
A. *Batch Learning (DS and BPS).*



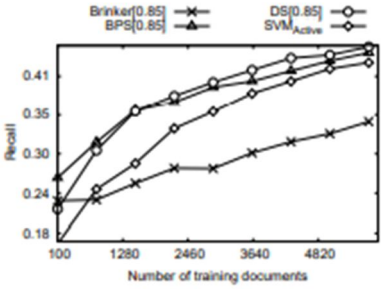
(a) Recall (D1)



(b) Recall (D2)



(c) Recall (D3)



(d) Recall (D4)

©IJRASET: All Rights are Reserved

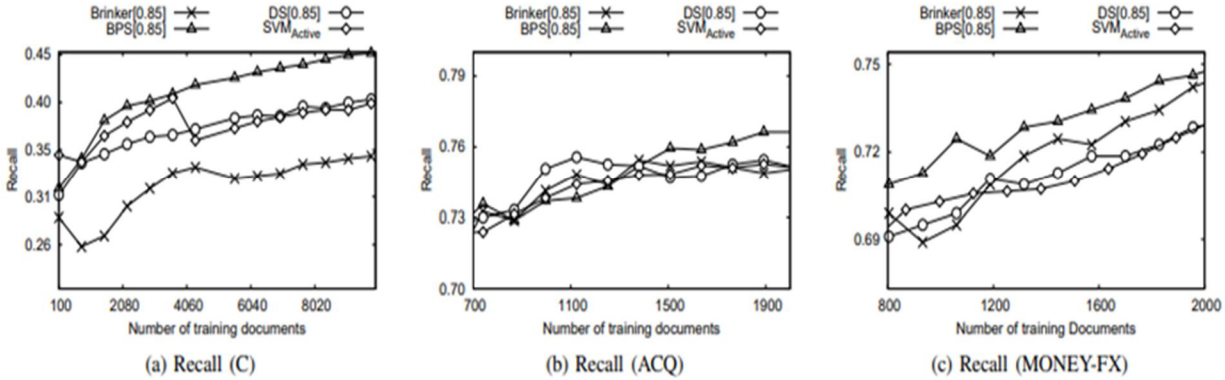


Figure 2: Recall of the algorithms for legal matters

As we can see from these charts, increasing the training data increases the model's recall, which is predicted for all techniques and datasets. BPS had the greatest recall of the compared to other methods. DS is the second-best approach overall, and its performance is excellent. For D2, D3, and D4 datasets, it's performance is nearly identical to BPS. With the exception of the D1 dataset, Brinker technique performance is worse for all of the proprietary datasets. SVMactive's performance is in between Brinker's and the proposed techniques.

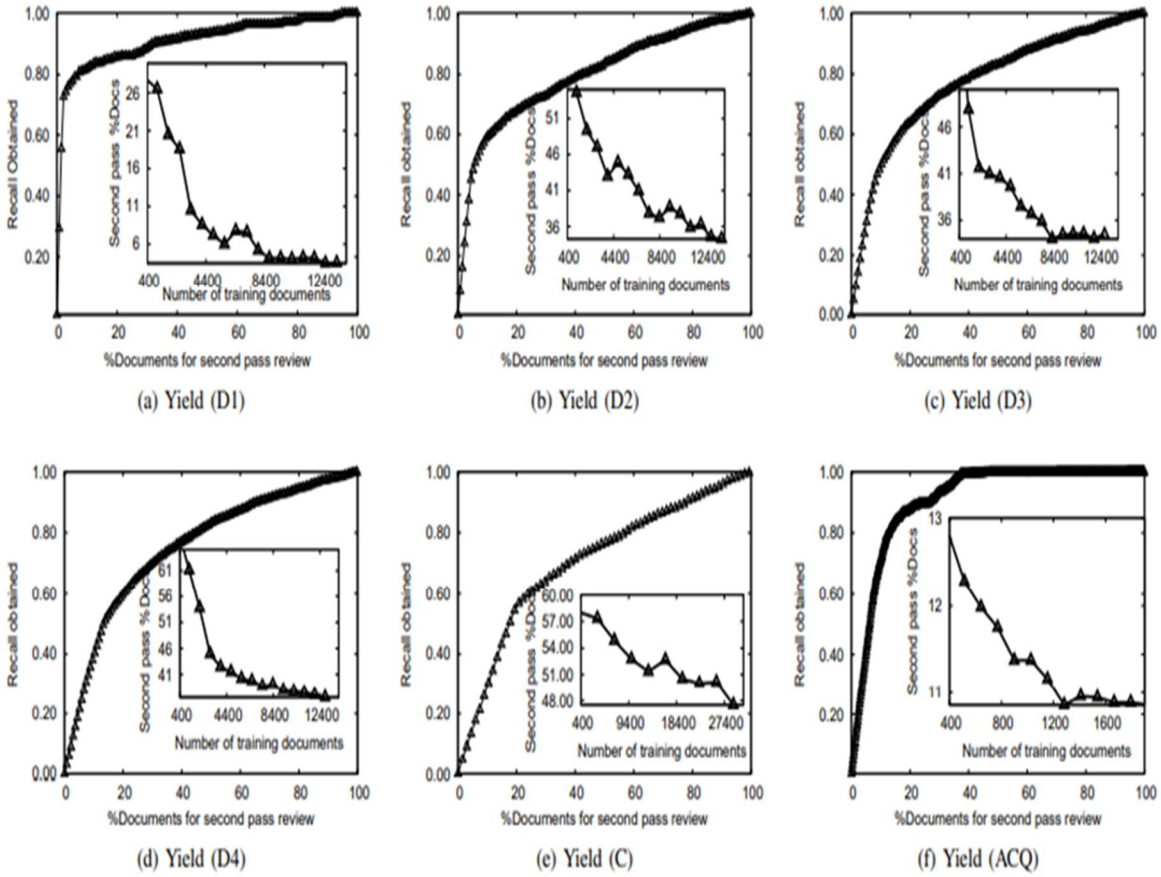


Figure 3: Yield Curve

The yield curve depicts the link between recollection and the minimal number of documents that must be read in order to attain that recall. The better the model and the fewer papers required for the second pass inspection, the steeper the yield curve.

©IJRASET: All Rights are Reserved

B. Explainable predictive coding using Logistic Regression.

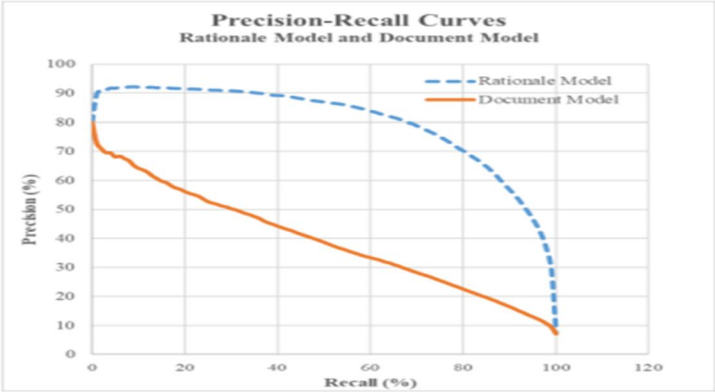


Figure 4: Precision-Recall Curves for Rationale Model and Document Model

Figure demonstrates the accuracy and recall curves for discriminating annotated responsive rationales from non-responsive text fragments using the document and reasoning models. Figure shows how well the reasoning models functioned. The reasoning models were 70% accurate with an 80% recall rate. The document models' performance was encouraging, despite being less successful than the logic models, especially given the 6.5 percent responsive document rate. The accuracy of the document models was more than 25% at 75% recall, which is roughly four times greater than the response rate.

Table VI. Rationale Recall for Rationale And Document Models

No. of words in snippet	Top K snippets	Rationale Recall	
		Rationale Model	Document Model
50	1	48%	44%
	2	62%	56%
	3	71%	65%
	4	76%	71%
	5	79%	75%
100	1	47%	51%
	2	64%	64%
	3	73%	73%
	4	79%	78%
	5	82%	82%
200	1	45%	60%
	2	68%	73%
	3	79%	81%
	4	84%	86%
	5	88%	89%

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue XI Nov 2021- Available at www.ijraset.com

Table illustrates the recall (or the proportion of responsive annotated rationales properly detected) of the document and reason models using different text snippet sizes. The reasoning models outperformed the document models for snippets of 50 words. Both model types demonstrated identical recall for snippets of 100 words, however the document models beat the reasoning models for snippets of 200 words.

A snippet almost always includes terminology not found in the annotated rationale, and it seldom spans the entire annotated justification. Noise was difficult to tolerate for reasoning models because they were trained using annotated rationales (irrelevant words).

Document models, on the other hand, were trained using the complete document text, which includes words from both the annotated rationales and the rest of the document, allowing them to be more noise tolerant.

Attorneys using the Rationale model to analyse the top four 50-word snippets (125 to 250 words, taking into account snippet overlap) could save 720 to 845 words on average from each text while maintaining a 76 percent recall rate. An attorney who uses the Document Model to analyse the initial top 50-word sample can cut 920 words from each document review while still getting a 44 percent recall rate.

C. CNN Parameters.

1) Impact of word representation

Table VII
Precision At 90% Recall For Various Word Representations

Word Representations	Precision on D1(%)	Precision on D2(%)	Precision on D3(%)
Dynamic Pre-trained WE	83.45	65.35	34.85
No Pre-trained WE	84.59	67.08	34.21
Static Pre-trained WE	82.86	60.73	33.69

2) Impact of kernel filter size

Table VIII
Precision At 90% Recall For Single Kernel Filter

Kernel Size	Precision on D1(%)	Precision on D2(%)	Precision on D3(%)
2	85.02	62.03	35.03
3	83.99	66.77	34.52
8	83.45	65.35	34.85
15	83.27	61.78	35.69
25	81.38	57.99	35.56
30	82.26	58.79	35.03

©IJRASET: All Rights are Reserved

Table IX
Precision At 90% Recall For Multiple Kernel Filter

Multiple Kernel Size	Precision on D1(%)	Precision on D2(%)	Precision on D3(%)
(1,2,3)	86.02	67.72	36.36
(4,5,6)	86.44	69.48	41.08
(8,8)	85.38	72.45	34.87
(9,10,11)	84.50	68.27	38.66
(11,12,13)	84.25	64.76	41.00

Table X
Precision Comparison At 90% Between Single And Multiple Kernel

Kernel	Precision on D1(%)	Precision on D2(%)	Precision on D3(%)
Best Single	85.02	66.77	35.69
Best Multiple	86.44	72.45	41.00

3) Impact of Choosing Number of Filters and Features Per Filter

Table XI
Precision AT 75% Recall

Number of filters	Feature per filter	Precision on D1(%)	Precision on D2(%)	Precision on D3(%)
256	1	92.83	88.56	51.11
64	4	89.50	79.91	44.20
16	16	85.59	51.43	30.19
4	64	79.47	29.86	27.78

4) Impact of Number of Filters

Table XII
Precision AT 75% Recall for Number of Filter

Number of filters	Precision on D1(%)	Precision on D2(%)	Precision on D3(%)
32	89.08	75.00	40.09
128	91.76	83.96	48.68
1024	92.64	90.86	52.15
2048	93.28	90.36	53.49

©IJRASET: All Rights are Reserved

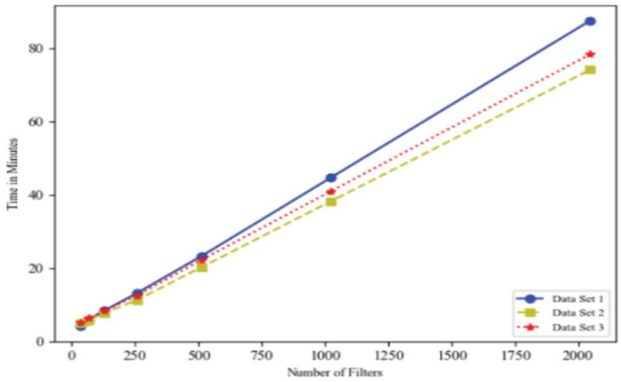


Figure 5: Computation Time

D. Active Learning.

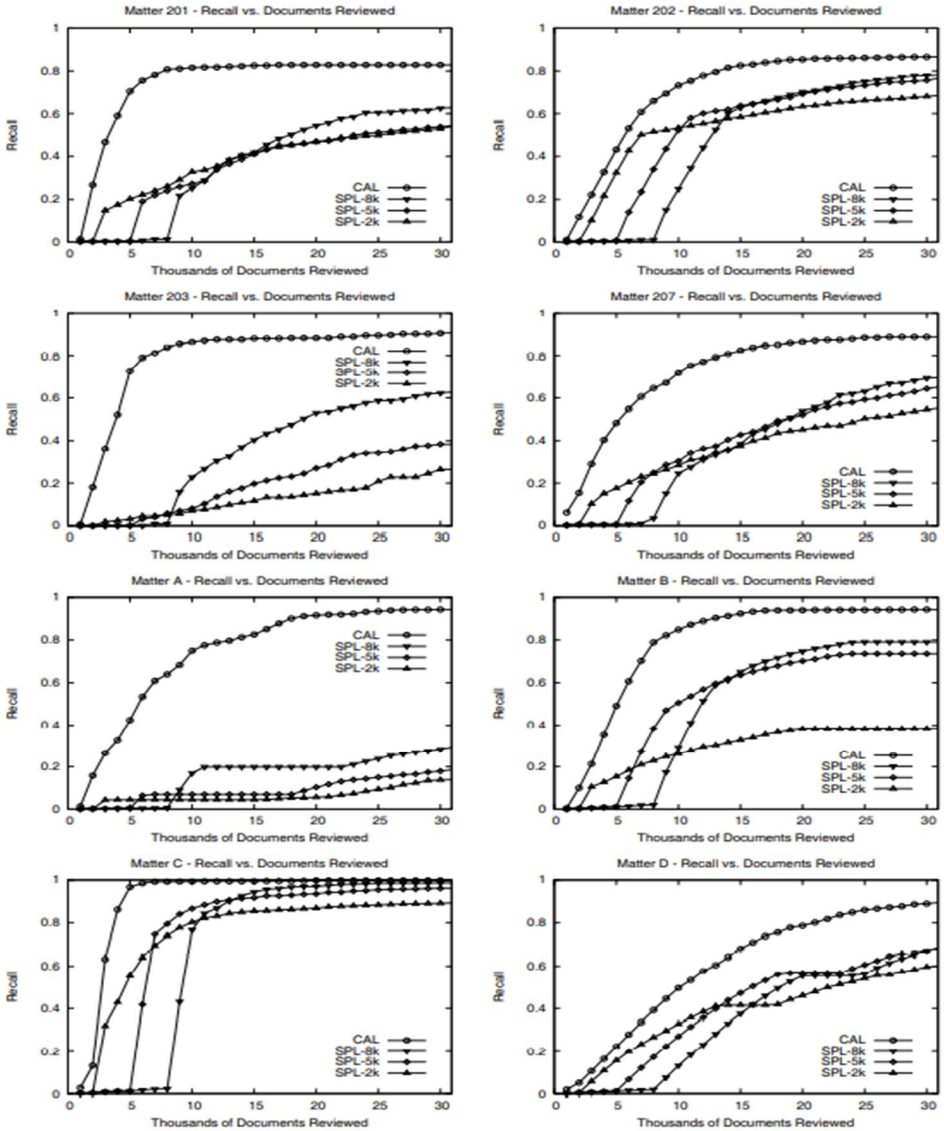


Figure 6: CAL vs SPL using three different using three different training-set sizes of randomly selected documents

©IJRASET: All Rights are Reserved

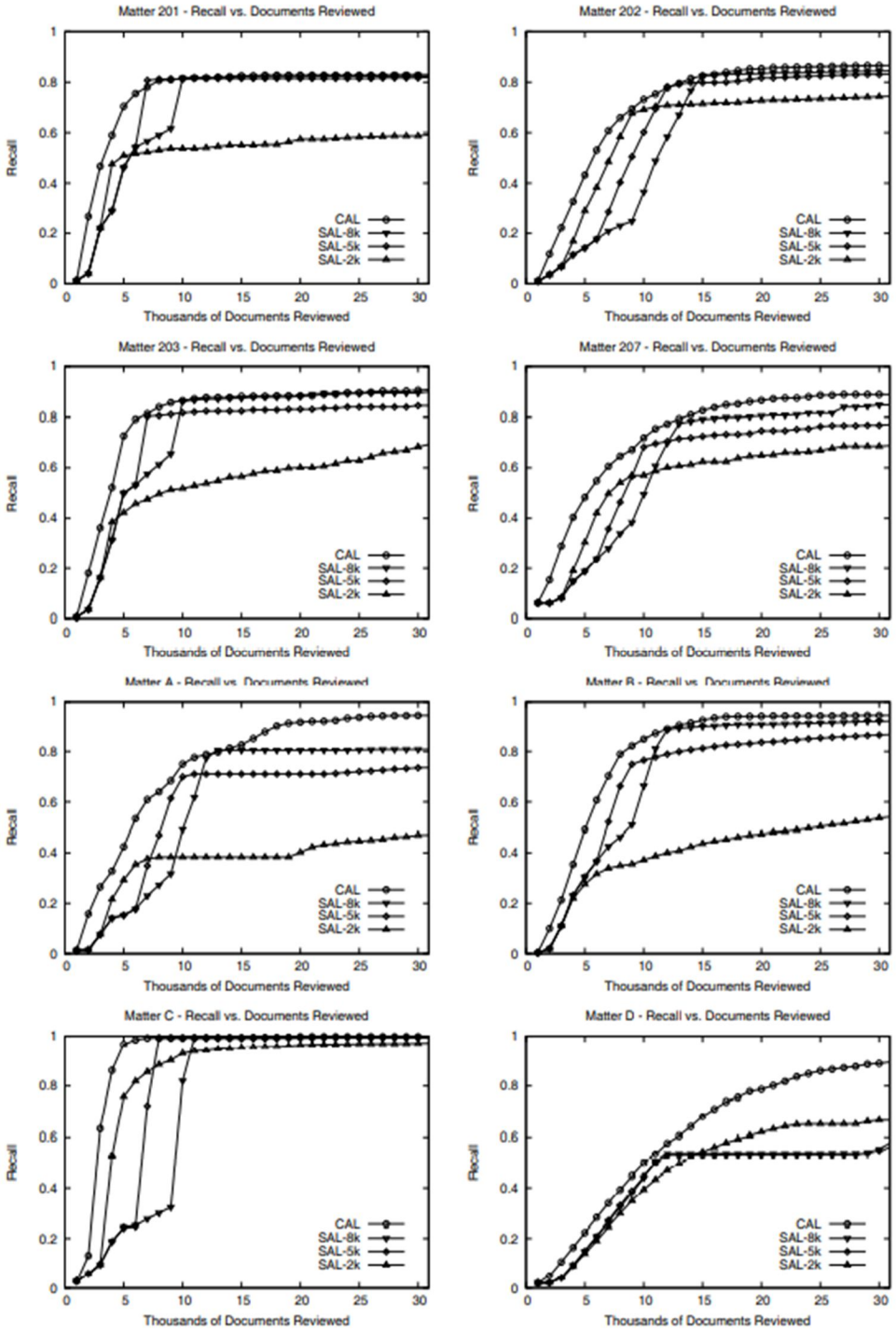


Figure 7: CAL vs SAL using three different training-set sizes of uncertainty-sampled documents.

For all relevant training-set sizes, the CAL technique yields stronger recall than SPL with less effort. The basic result is the same in all eight graphs: The CAL curve indicates a high slope after the first 1,000 documents (i.e., the seed set) that is maintained until the bulk of relevant documents have been found. The slope begins to level down dramatically about 70 percent recollection and essentially plateaus between 80 percent and 100 percent recall. The SPL curve has a low slope during the training phase, a high slope during the review phase, a falloff, and ultimately a plateau. The CAL procedure has a greater recall rate than the SAL protocol. Unlike the SPL gain curves, the SAL gain curves frequently meet the CAL curves at a single inflection point.

©IJRASET: All Rights are Reserved

E. Unscented Kalman Filtering.

Validation AUC and Recall at 500,750 and 1000 documents reviewed with active label request compared with n=280

Table XIII
Results

MODEL	AVERAGE VALIDATION AUC		AVERAGE VALIDATION RECALL AT 500 DOCS		AVERAGE VALIDATION RECALL AT 1000 DOCS	
Number of Documents Reviewed per Round	n=280	Active Label Requ,est	n=280	Active Label Request	n=280	Active Label Request
UKF + RANDOM	87%	86%	67%	67%	92%	92%
UKF + DECISION VALUE	87%	86%	69%	67%	92%	92%
UKF + TOP RANKED	87%	87%	69%	67%	93%	93%

V. CONCLUSION

In this paper, we analysed five predictive coding algorithms and methods. These are the algorithms and methods that we plan to implement in our model of using predictive coding and TAR for document review (specifically pdf, excel sheets and text docs) to finally conclude the most effective and efficient one.

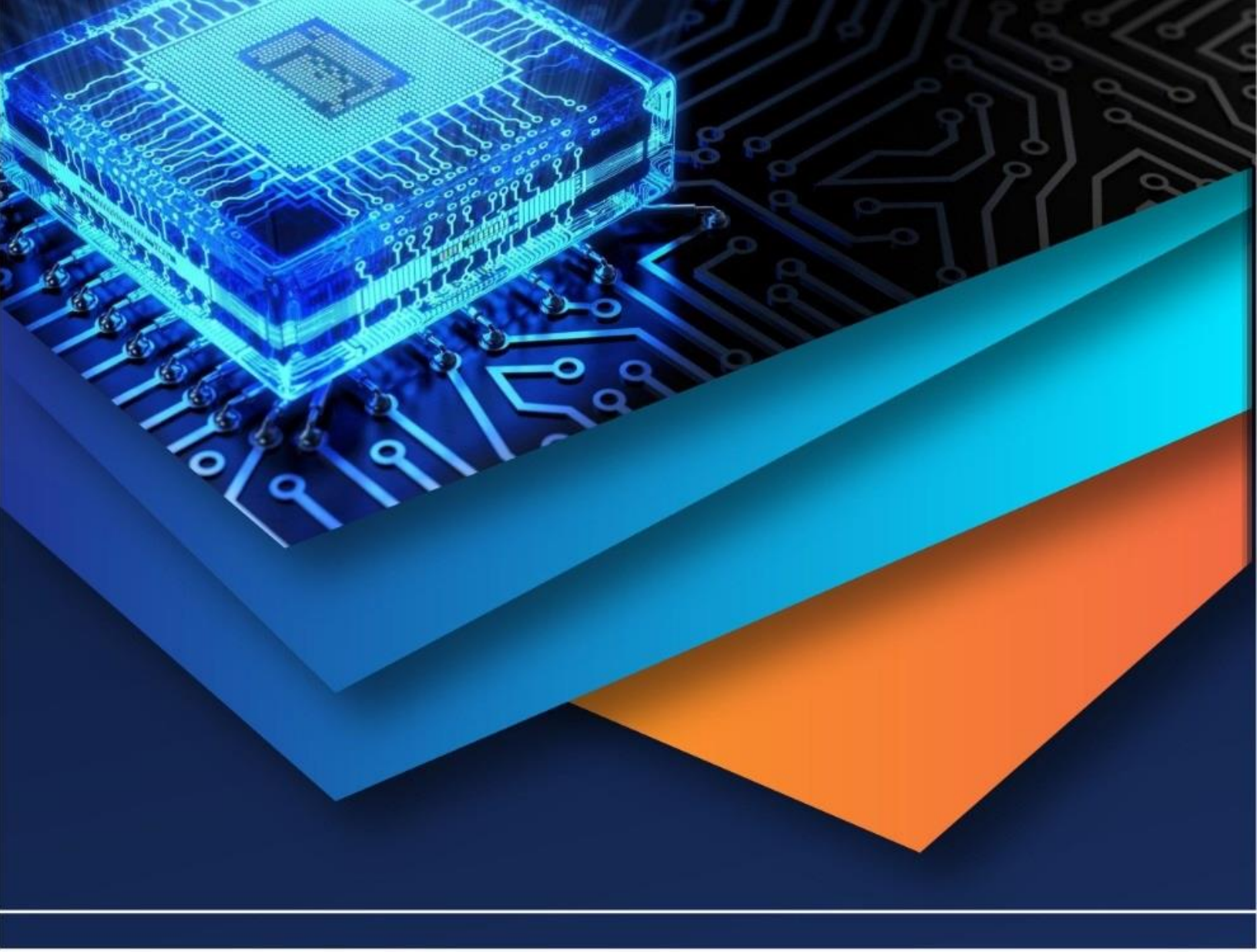
VI. ACKNOWLEDGEMENT

The authors of this paper would like to thank Mrs. Snehal Shintre for her support and guidance in making this work possible.

REFERENCES

[1] Yihua Shi Astle, Xuning Tang, Craig Freeman, “Application of Dynamic Logistic Regression with Unscented Kalman Filter in Predictive Coding”, IEEE International Conference on Big Data, 2017 .
[2] Tanay Kumar Saha, Mohammad Hasan, “Batch-mode Active Learning for Technology-Assisted Review”, IEEE Big Data, 2015.
[3] Rishi Chhatwal, Peter Gronvall, Nathaniel Huber-Fliflet, “Explainable Text Classification in Legal Document Review (A Case Study of Explainable Predictive Coding)”, IEEE International Conference on Big Data (Big Data), 2018.
[4] Gordon V. Cormack, Maura R. Grossman, “Evaluation of Machine-Learning Protocols for Technology-Assisted Review in Electronic Discovery”.
[5] Qian Han, Yufeng Kou, Derek Snaidauf, “Experimental Evaluation of CNN Parameters for Text Categorization in Legal Document Review”, IEEE International Conference on Big Data (Big Data), 2019.
[6] Christian J. Mahoney, Nathaniel Huber-Fliflet, Katie Jensen, “Empirical Evaluations of Seed Set Selection Strategies for Predictive Coding”, IEEE International Conference on Big Data, 2018.
[7] Article on “ProSearch-TAR-Solutions-for-a-New-Decade”, by ProSearch https://www.prosearch.com/wp-content/uploads/2021/01/ProSearch-TAR-Solutions - for-a-New-Decade.pdf.
[8] Christian J. Mahoney, Jianping Zhang, Nathaniel Huber-Fliflet, Peter Gronvall, Haozhen Zhao, “A Framework for Explainable Text Classification in Legal Document Review”, IEEE International Conference on Big Data (Big Data), 2019.
[9] Sauvola J., Pietikznen M., and Koivusaari M., “Predictive Coding for Document Layout Characterization”, DAPMachine Vision and Media Processing Group, Infotech Oulu, University of Oulu, Finland, 1997.
[10] M. W. Spratling, “Predictive Coding as a Model of Cognition”, King’s College London, Department of Informatics, London, UK, 2017.
[11] Beren Millidge, Anil K Seth, Christopher L Buckley, “PREDICTIVE CODING: A THEORETICAL AND EXPERIMENTAL REVIEW”, 2021.
[12] Rishi Chhatwal, Nathaniel Huber-Fliflet, Robert Keeling, Dr. Jianping Zhang, Dr. Haozhen Zhao, “Empirical Evaluations of Preprocessing Parameters’ Impact on Predictive Coding’s Effectiveness”, IEEE International Conference on Big Data (Big Data), 2016.
[13] Rishi Chhatwal, Nathaniel Huber-Fliflet, Robert Keeling, Dr. Jianping Zhang, Dr. Haozhen Zhao, “Empirical Evaluations of Active Learning Strategies in Legal Document Review”, IEEE International Conference on Big Data (BIG DATA), 2017.
[14] Peter Gronvall, Nathaniel Huber-Fliflet, Dr. Jianping Zhang, Robert Keeling, Robert Neary, Dr. Haozhen Zhao, “An Empirical Study of the Application of Machine Learning and Keyword Terms Methodologies to Privilege-Document Review Projects in Legal Matters”, IEEE International Conference on Big Data (Big Data), 2018.
[15] Michael Spratling, “A Review of Predictive Coding Algorithms”, King’s College London, Department of Informatics, London, UK, 2016.
[16] Gordon V. Cormack, Maura R. Grossman, “Evaluation of Machine-Learning Protocols for Technology-Assisted Review in Electronic Discovery”.
[17] Gideon Christian, “PREDICTIVE CODING: ADOPTING AND ADAPTING ARTIFICIAL INTELLIGENCE IN CIVIL LITIGATION”, 2019.

©IJRASET: All Rights are Reserved



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)