

RESEARCH ARTICLE

A Representation-Based Query Strategy to Derive Qualitative Features for Improved Churn Prediction

SOUMI DE¹, (Member, IEEE), **AND P. PRABU²**¹Department of Data Science, CHRIST (Deemed to be University), Bengaluru 560029, India²Department of Computer Science, CHRIST (Deemed to be University), Bengaluru 560029, India

Corresponding authors: Soumi De (soumi.de@res.christuniversity.in) and P. Prabu (prabu.p@christuniversity.in)


ABSTRACT The effectiveness of any Machine Learning process depends on the accuracy of annotated data that is used to train a learner. However, manual annotation is expensive. Hence, researchers adopt a semi-supervised approach called active learning that aims to achieve state-of-the-art performance using minimal number of samples. Although it boosts classifier performance, the underlying query strategies are unable to eliminate redundancy in selected samples. Redundant samples lead to increased cost and sub-optimal performance of learner. Inspired by this challenge, the study proposes a new representation-based query strategy that selects highly informative and representative subsets of samples for manual annotation. Data comprises messages of a set of customers sent to a service provider. Series of experiments are conducted to analyze the effectiveness of the proposed query strategy, called “Entropy-based Min Max Similarity” (E-MMSIM), in the context of topic classification for churn prediction. The foundation of E-MMSIM is an algorithm that is popularly used to sequence proteins in protein databases. The algorithm is modified and utilized to select the most representative and informative samples. The performance is evaluated using F1-score, AUC and accuracy. It is observed that “E-MMSIM” outperforms popular query strategies, and improves performance of topic classifiers for each of the 4 topics of churn prediction. The trained topic classifiers are used to derive qualitative features. These features are further integrated with structured variables for the same group of customers to predict churn. Experiments provide evidence that inclusion of qualitative features derived using E-MMSIM, enhance the performance of churn classifiers by 5%.

INDEX TERMS Active learning, churn prediction, query strategy, entropy, topic classification.

I. INTRODUCTION

Machine learning (ML) has emerged as a powerful means to understand the hidden pattern in data. The core of any ML workflow lies in the quality and quantity of the labeled data that is used to train the learning models. However, in cost-sensitive circumstances, acquiring labeled data can be quite challenging. A semi-supervised approach called active learning (AL) is often used to address the problem of cost associated with labeled data [1], [2]. AL initiates algorithmic training with only few annotated samples. Thereafter, small

batches of most informative unlabeled instances are selected for manual annotation, and are included as part of the training set. The process of training is repeated until a pre-defined level of model performance is achieved, or the limit for the maximum number instances that can be considered for manual annotation, is reached. The selection of fixed number of unlabeled samples is primarily driven by query strategies that are often driven by posterior probability output of a classifier. Query strategies (QS) aim to improve model performance with each iteration of sample selection. Although QS is useful in optimizing the samples selected for manual labeling, they are limited by the problem of redundancy in the shortlisted samples. Redundant samples

The associate editor coordinating the review of this manuscript and approving it for publication was Alberto Cano .

that are selected for manual annotation leads to increased time and resource consumption. As a result, further research is needed to address the problem.

This paper quantifies the impact of commonly used QS on the performance of a classifier. Experiments are conducted that highlight the problem of redundancy in AL. A new QS called “Entropy-based Min Max Similarity” (E-MMSIM) is proposed that attempts to tackle this drawback by combining similarity profiles of samples with mis-classification cost of a classifier. Results are positive and show that the newly proposed QS leads to enhanced performance of classifiers in AL. The remaining paper includes Section 2 that presents related work on various QS approaches frequently used in AL. Section 3 highlights the methodology used in the study. Section 4 provides an outline of the model evaluation measures. Results are discussed in section 5. Finally, section 6 contains the conclusion.

II. RELATED WORK

A. ACTIVE LEARNING (AL)

AL seeks to optimize a classifier performance using minimum number of labeled instances. As shown in Fig. 1, a typical AL workflow consists of a labeled dataset L for training and an unlabeled pool U . Each iteration of the workflow involves instances of U , denoted by x_U , to be annotated by oracle, and added back to L for training the classifier. An ideal solution to achieve the best performing classifier is to have oracle label every element of U . However, this is not a cost-effective approach. Hence, the goal of AL is to use query strategies to find an optimized training subset that enables a classifier achieve equivalent performance compared to a classifier that is trained on the full annotated set [3], [4].

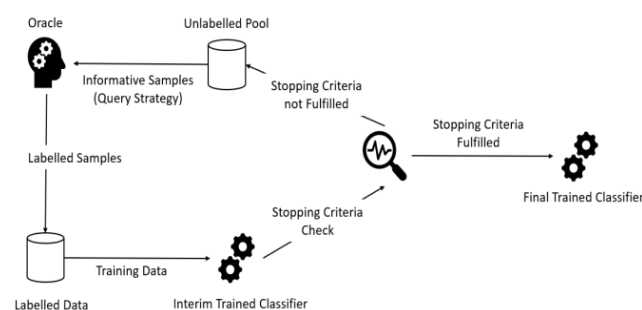


FIGURE 1. Active learning workflow.

B. QUERY STRATEGIES

Query strategies lie at the core of AL workflow. They define the measure that is used to quantify the informativeness of a sample in the unlabeled pool. There are many types of query strategies that can be used in AL.

This paper primarily focuses on posterior probability-based query strategies that utilize pool-based sampling to select the most informative samples for oracle to label. The

underlying assumption is that most informative samples are those for which the classifier is most uncertain. Hence, pool-based sampling is also known as uncertainty sampling. Uncertainty sampling calculates the uncertainty measure for each sample in a large pool of unlabeled dataset U . Thereafter, the top N most uncertain instances from U are selected for manual annotation. There are 2 main measures that are frequently used to evaluate the informativeness or uncertainty of a sample – entropy, and least confidence measure.

1) ENTROPY SAMPLING

Entropy-based uncertainty sampling uses entropy as a measure for informativeness. Entropy E is defined as the degree of uncertainty in an instance

$$E = - \sum p_i \times \log(p_i) \quad (1)$$

where i stands for the number of classes, and p_i represents the probability distribution for the i^{th} class. Samples from unlabeled pool are ranked in descending order of entropy, and instances with highest entropy are selected for annotation [5], [6]. A drawback of entropy-based sampling is that the method is unable to eliminate redundant samples being selected for annotation in each iteration of AL. This causes wastage of resources and more time required for AL workflow to converge.

2) LEAST CONFIDENCE SAMPLING

Least confidence (LC) uncertainty sampling is a method that is driven by the minimum class probability measure. In case of binary classification, the measure is defined as

$$LC = 1 - p_i \quad (2)$$

where p_i is the probability distribution of the most confident class. Each instance from unlabeled pool is ranked in descending order of LC. Samples with highest LC are selected for manual annotation [7], [8], [9]. Although intuitive, redundancy in selected samples is a major drawback of this method. Additionally, LC is not recommended in multi-class setting.

Using aforementioned query functions, the most informative samples are shortlisted and annotated in each iteration of AL. Recent literature has many use cases where these strategies are extensively used. The next section discusses few recent works in this field.

A Cost Effective Active Learning (CEAL) method is proposed in [10]. The study initializes the learning parameters of Convolutional Neural Network (CNN) model by utilizing the initial set of labeled samples. Both entropy and least confidence sampling are used for selecting informative samples. The authors additionally utilize most confident samples from the pool and add them to the labeled dataset for training the classifier and updating the learning parameters. Although the proposed method achieves maximum accuracy by utilizing 60% of labeled data on two image data sets, the advantage gained by including most confident samples in training set, needs more analysis and experimentation.

Another study utilizes 3 classification models called CNN, Support Vector Machines (SVM) and Random Forest (RF), for categorization of medical images [11]. LC sampling is used to select the most uncertain pixels for annotation. CNN outperforms other classifiers achieving highest accuracy with a smaller number of AL iterations. The study compares performance of 3 models on 2 cancer-related image datasets. However, the problem of redundancy in the selected pixels, is neither discussed, nor highlighted.

CNN is used as a classifier in another image segmentation task that runs AL iterations with ensembles [12]. The study uses ensembles to derive the uncertainty estimates of unlabeled data with entropy-based query strategy as one of the acquisition functions to select the samples for annotation. The AL strategies outperform their standalone model uncertainty estimates. Although the problem of redundancy in entropy-based query strategy is not discussed explicitly, the authors attempt to incorporate representativeness in selected samples using REPR strategy as a separate experiment. The drawback of ensemble-based approaches being computationally expensive is accepted by authors.

Another study combines geometric smoothness priors in the image space with traditional entropy-based informativeness measures to estimate pixels that are most informative for annotation [13]. The study is performed on an image dataset under a multi-class setting and exploits geometric properties of data for selection of the most informative pixels. However, the authors acknowledge the limitation of explicitly designing hand crafted query functions for every new use case under consideration.

As described earlier in the section, both entropy and least confidence query functions are limited by redundancy in selected samples. In both the query strategies, the focus is limited to a small region of data distribution which may lead to increase in the bias of the classifier. It is found that solutions that aim to reduce the above-mentioned limitation in selection strategies of AL, are limited in literature.

III. METHODOLOGY

A. DATASET

There are 2 types of data sets used in this study. The first data set is qualitative in nature and comprises of textual interaction messages of 20,300 customers with a service provider for the period 2020 to 2021. The service provider for this study is a renowned hotel commerce platform company. The qualitative data is divided into 4 smaller textual sets, each meant to train a classifier on a particular topic related to churn prediction. The 4 datasets, denoted by D1, D2, D3 and D4, will be considered for training topic classifiers to detect the presence or absence of a topic in a given message. Each dataset contains 24,000 labeled messages. The data is preprocessed by following standard process of removing stop words, punctuation, contractions, special characters and hyperlinks. Emoticons are converted to text for gaining better context of emotion. This is finally followed by lemmatization.

The presence and absence of a topic in a message is depicted by positive and negative class respectively.

The second data set is structured in nature. It contains 78 quantitative independent variables that are commonly used for churn prediction for the same group of 20,300 customers, each row, representing a customer. The independent variables are related to demography and the product usage features of a user for the year 2020-2021. The dependent variable is a binary attribute that distinguishes a churner from a non-churner. The dataset has a class distribution of 12% that represent customers who have churned, or terminated the contract with the service provider.

B. PROPOSED QUERY STRATEGY (E-MMSIM)

The study integrates a new representation-based query strategy E-MMSIM within AL framework in the context of topic classification for churn prediction. E-MMSIM exploits Hobohm-1 algorithm that is frequently used in the domain of protein sequencing. The algorithm is further extended to derive a new query function that gives importance to both informativeness and representativeness of a sample. Most studies that attempt to incorporate representativeness in query strategies are based on clustering methods that are computationally inefficient and require pre-defined parameters as input. However, E-MMSIM utilizes computationally fast Hobohm-1 algorithm and does not need clustering. The approach makes E-MMSIM easy to implement on large datasets, and at the same time eliminates redundancy from the selected samples.

1) HOBOHM-1 ALGORITHM

Protein data banks store information related to three-dimensional coordinates of protein sequences. There is considerable redundancy in protein databases where many protein sequences are similar to each other, and ultimately render the database impracticable for statistical analyses. In 1992, authors Hobohm et al. proposed a simple, fast and effective protocol (algorithm 1) to overcome the problem of redundancy in protein sequencing database [14]. Fig. 2 presents the steps involved in Hobohm-1 algorithm.

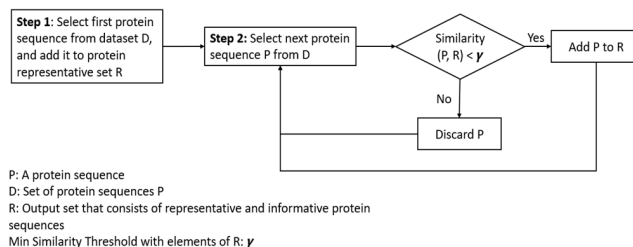


FIGURE 2. Hobohm-1 Algorithm.

The algorithm processes each protein sequence P from a sorted list of candidate proteins by iteratively discarding the sequences that are similar to the already selected proteins in R. In Fig. 2, the similarity threshold used to either select or

discard a protein sequence is depicted by γ . Mathematically, the algorithm optimizes the following function:

$$\begin{aligned} & \text{maximize } |R|_{R \subseteq P} \text{ such that } \text{similarity}(p_1, p_2) < \gamma \\ & \text{for all } p_1, p_2 \in R \end{aligned} \quad (3)$$

2) E-MMSIM

E-MMSIM is based on a tripartite hierarchical approach that integrates previously described informativeness measure, called entropy, with an extended and modified version of Hobohm-1 algorithm that considers mis-classification similarity. As shown in Fig. 3, E-MMSIM incorporates 3 essential features to tackle redundancy in selected samples. Firstly, the algorithm considers the ‘‘Entropy Criterion’’ by sorting the unlabeled samples in descending order of entropy. This makes certain that the algorithm targets those samples that have maximum uncertainty. Secondly, the algorithm optimizes the representativeness of each selected sample of the unlabeled population. This property is called the ‘‘Representative Criterion’’ and is integrated in E-MMSIM by means of the Hobohm-1 algorithm. The study further extends the Hobohm-1 algorithm to integrate the mis-classification similarity between selected samples and mis-classified samples of test set. This is denoted by the ‘‘Informativeness Criterion’’. The rationale behind incorporating mis-classification similarity in the proposed query strategy is mainly to optimize the informativeness of a sample. This ensures that the final batch of selected samples have some level of similarity with misclassified samples in test set that are not well represented in the training set.

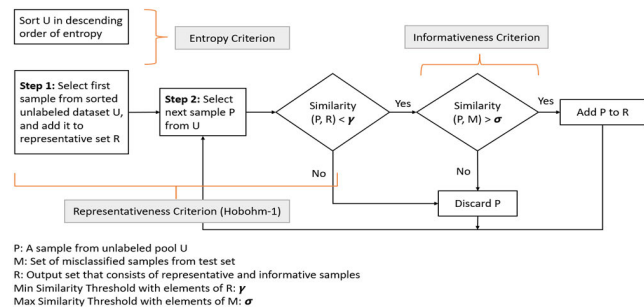


FIGURE 3. E-MMSIM.

The algorithm begins by splitting the 24,000 labelled instances in a dataset into training set D_{train} , test set D_{test} , and pool U. D_{train} has 500 randomly selected labeled samples, while D_{test} consists of 5,000 samples. The remaining 18,500 instances are moved to the pool U. The class distribution is maintained at 50%. A binary classifier is trained using D_{train} and is used for classifying instances in D_{test} and pool U. Let M denote the set of all the misclassified instances in D_{test} and R represent final representative list of instances for annotation. Initially, R is an empty set. Thereafter, entropy is calculated for each sample in U and a sorted list in descending order of entropy is prepared. The instances on the top of this sorted list depict the ones with highest uncertainty. These instances

are iteratively selected and added to the representative set R if they satisfy 2 conditions related to similarity thresholds. First condition is that the cosine similarity between the selected sample of U and all the elements in R, should be below γ . Second condition is that the cosine similarity between selected sample of U and all the elements in M, should be above similarity threshold σ . The rationale behind introducing the two similarity thresholds is that while condition 1 optimizes the representativeness criterion of Hobohm-1 algorithm using γ , condition 2 optimizes the informativeness criterion using σ . In the study, the optimal values for γ and σ are found to be 0.3 and 0.6 respectively. Elements that satisfy the above 2 conditions are added to R, and the process is repeated until the number of elements in R reaches a pre-defined batch size. R is the set of samples that are selected for annotation and is added to D_{train} . This completes a single iteration of AL. The process is repeated for a fixed number of iterations for each of the 4 datasets. Performance of the trained classifier on D_{test} is recorded in each iteration. Mathematically, E-MMSIM can be represented by (4):

$$\begin{aligned} & \text{maximize } |R|_{R \subseteq U} \text{ such that } \text{sim}(u_R, u) < \gamma \\ & \text{for all } u_R \in R, u \in U \\ & + \text{maximize } |I|_{R \subseteq U} \text{ such that } \text{sim}(u, m) > \sigma \\ & \text{for all } m \in M, u \in U \end{aligned} \quad (4)$$

IV. EVALUATION MEASURE

The study uses 3 parameters to evaluate and compare the performance of E-MMSIM-based sampling with traditional methods of entropy and least confidence-based sampling that were earlier discussed in section 2. The objective is to train a binary classifier to detect the presence or absence of a topic. As discussed before, the presence of a topic represents a positive class and absence of a topic represents a negative class. The following section discusses each evaluation metric in detail.

A. F1-SCORE

F1-score is the geometric mean of precision and recall [15], [16], [17]. Precision is defined as the proportion of correctly classified positive instances. It is derived using (5). On the other hand, recall stands for the proportion of positive instances that are classified correctly. It is calculated using (6). F1-score can then be derived using (7).

$$\text{Precision} = \frac{TP}{(TP + FP)} \quad (5)$$

$$\text{Recall} = \frac{TP}{(TP + FN)} \quad (6)$$

$$F1 - \text{score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{(\text{Precision} + \text{Recall})} \quad (7)$$

In the above equations, True Positive (TP) is the total number of positive instances that are correctly classified. False Positive (FP) represents number of negative instances that are

incorrectly classified. Lastly, False Negative (FN) represents the number of incorrectly classified negative samples.

B. AREA UNDER CURVE (AUC)

AUC is an evaluation parameter that is independent of any decision threshold used for classification. It is defined as the area under the Receiver Operating Curve (ROC). The AUC value is an integral of ROC that plots True Positive Rate (TPR) against False Positive Rate (FPR) [18]. AUC has a range between 0 and 1. A good classifier has AUC that is close to 1.

$$AUC = \int TPR d(FPR) \quad (8)$$

C. ACCURACY

Accuracy is one of the most widely used evaluation parameters. It is defined as the proportion of the total correctly classified instances [19]. It is derived mathematically, using the following equation:

$$Accuracy = \frac{TP + TN}{n} \quad (9)$$

where TN stands for True Negative that represents the number of correctly classified negative samples, and n denotes the total number of samples.

V. RESULTS AND DISCUSSION

To evaluate the performance gain achieved by using E-MMSIM, 4 experiments are performed independently. In all the experiments, the number of AL iterations, represented by k, are limited to 40, with a batch size b of 25 samples to be selected in each iteration. The budget B is set to 1,500. The dataset is split into training set D_{train} , test set D_{test} and pool U, and a vectorizing function V is applied to convert the datasets into numerical format. The vectorized datasets are represented by D''_{train} , D''_{test} and U'' respectively. XGBoost is used as a binary classifier. It is denoted by C in each experiment. The algorithms in this study utilize Python with Spyder as the editing environment. Spyder is an open-source development environment that is available in Anaconda. Operating system used is Windows 11 with Intel Core i7-10510U, 2.3 GHz processor. The machine has a 16GB RAM that is utilized for conducting the experiments.

A. EXPERIMENT 1: RANDOM SAMPLING (RS)

The first experiment is an attempt to establish a baseline. It records the performance on D_{test} when no query strategy is applied and samples are selected randomly in each iteration of the AL process. The pseudo-code of this experiment is presented in Fig. 4.

In this experiment, classifier C is trained using D''_{train} . Performance of trained classifier C on D''_{test} is recorded and added to set P. Thereafter, b number of samples are selected randomly. This batch of selected samples is represented by R. The samples in R are added to D_{train} and removed from

Algorithm: Topic Classification with Active Learning using Random Sampling

Input:

Data set $D_{train} = \{tr_1, tr_2, \dots, tr_p\}$, $D_{test} = \{ts_1, ts_2, \dots, ts_q\}$, $U = \{u_1, u_2, \dots, u_j\}$,
Batch size: b
Number of active learning iterations: k
Vectorizing function V
Classification algorithm C

Process:

$R = \{\}$ // List of selected samples for annotation
 $P = \{\}$ // List of performance recorded at each iteration
for $i = 1 \dots k$:
// Vectorize D_{train} , D_{test} and U
 $D''_{train} = V(D_{train})$
 $D''_{test} = V(D_{test})$
 $U'' = V(U)$

Train classifier C using D''_{train}
 $P = P + P_i$ // Record performance on D''_{test} for i^{th} iteration
 $R = \{u_1, u_2, \dots, u_b\}$ // Randomly select b samples from U
 $D_{train} = D_{train} + R$
 $U = U - R$

Output:

Performance set P for active learning using Random Sampling

FIGURE 4. AL with random sampling for topic classification.

Algorithm: Topic Classification with Active Learning using Entropy Sampling

Input:

Data set $D_{train} = \{tr_1, tr_2, \dots, tr_p\}$, $D_{test} = \{ts_1, ts_2, \dots, ts_q\}$, $U = \{u_1, u_2, \dots, u_j\}$,
Batch size: b
Number of active learning iterations: k
Vectorizing function V
Classification algorithm C

Process:

$R = \{\}$ // List of selected samples for annotation
 $P = \{\}$ // List of performance recorded at each iteration
for $i = 1 \dots k$:
// Vectorize D_{train} , D_{test} and U
 $D''_{train} = V(D_{train})$
 $D''_{test} = V(D_{test})$
 $U'' = V(U)$

Train classifier C using D''_{train}
 $P = P + P_i$ // Record performance on D''_{test} for i^{th} iteration
 $U'' = \{u_1, u_2, \dots, u_j\}$ // Sorted list of U in descending order of entropy
 $R = \{u_1, u_2, \dots, u_b\}$ // Select top b samples from U''
 $D_{train} = D_{train} + R$
 $U = U - R$

Output:

Performance set P for active learning using Entropy Sampling

FIGURE 5. AL with entropy sampling for topic classification.

pool U. The process is repeated k times. In each iteration, a new batch of samples are moved from U to D_{train} .

B. EXPERIMENT 2: ENTROPY SAMPLING (ES)

The second experiment records the performance on D_{test} when samples are selected using the ‘‘sorted entropy’’ condition in each iteration of the AL process. The pseudo-code of this experiment is highlighted in Fig. 5.

In experiment 2, classifier C is trained using D''_{train} . Performance of trained classifier C on D''_{test} is recorded and added to set P. Entropy is calculated for each element of U''

```

Algorithm: Topic Classification with Active Learning using E-MMSIM

Input:
Data set  $D_{train} = \{tr_1, tr_2, \dots, tr_p\}$ ,  $D_{test} = \{ts_1, ts_2, \dots, ts_q\}$ ,  $U = \{u_1, u_2, \dots, u_j\}$ ,
Batch size:  $b$ 
Number of active learning iterations:  $k$ 
Vectorizing function  $V$ 
Classification algorithm  $C$ 
Representation similarity threshold:  $\gamma$ 
Misclassification similarity threshold:  $\sigma$ 

Process:
 $R = \{\}$  // List of selected samples for annotation
 $P = \{\}$  // List of performance recorded at each iteration
for  $i = 1 \dots k$ :
// Vectorize  $D_{train}$ ,  $D_{test}$  and  $U$ 
 $D'_{train} = V(D_{train})$ ;  $D'_{test} = V(D_{test})$ ;  $U' = V(U)$ 

Train classifier  $C$  using  $D'_{train}$ 
 $P = P + P_i$  // Record performance on  $D'_{test}$  for  $i^{th}$  iteration
 $U' = \{u_1, u_2, \dots, u_j\}$  // Sorted list of  $U$  in descending order of entropy
 $M = \{m_1, m_2, \dots, m_j\}$  // List of misclassified samples in  $D'_{test}$ 
 $R = \{u_i\}$ 
for  $j = 2 \dots r$ 
if  $\text{similarity}(R, u_j) < \gamma$  and  $\text{similarity}(M, u_j) > \sigma$ :
 $R = R + u_j$ 
if number of elements in  $R > b$ 
break
 $D_{train} = D_{train} + R$ 
 $U = U - R$ 

Output:
Performance set  $P$  for active learning using LC Sampling

```

FIGURE 6. AL with E-MMSIM sampling for topic classification.

and a sorted list of U' is prepared in descending order of entropy. Let R denote the top b samples with highest entropy in U' . The samples in R are added to D_{train} , and removed from pool U . The process is repeated k times. In each iteration, new set of samples are moved from U to D_{train} .

C. EXPERIMENT 3: LC SAMPLING

The third experiment aims to understand the effect of LC sampling on model performance. Classifier is evaluated on D_{test} where samples are selected using the sorted “least confidence” criterion in each cycle of AL. The experiment is similar to entropy sampling, with the exception that in this experiment, LC function is used instead of entropy.

D. EXPERIMENT 4: E-MMSIM SAMPLING

The final experiment incorporates E-MMSIM as a sampling strategy in the AL process flow. As discussed before, data set D_{test} is used for performance evaluation. In this case, samples are selected using the combined “entropy-representativeness-informativeness” criterion to study the performance gain achieved by the newly introduced query function. The pseudo-code of this experiment is presented in Fig. 6.

In this experiment, classifier C is trained using D'_{train} . Performance of trained classifier C on D'_{test} is recorded and added to set P . Entropy is derived for each element of U' and the set is sorted in descending order of entropy. Let M denote the set of the misclassified samples in D'_{test} , and R represent the selected batch of b samples to be added to D_{train} . R is initialized to an empty set. Top elements of sorted U' are selected and added to R , provided they fulfil 2

similarity criteria called the Min-Max SIMilarity (MMSIM) criteria. The MMSIM criteria involves 2 thresholds, γ and σ . Threshold γ is the maximum similarity between a sample in U' and each element of R . Threshold σ is the minimum similarity between a sample in U' and each element of M . Hence, if similarity of a sample in U' is less than γ for each element of R , and greater than σ for each element of M , the sample is added to R . The MMSIM criteria ensures that only the representative and informative samples are selected for annotation, thereby reducing redundancy in the set R . For experiment 4, through experimental analysis, it is found that the optimal values for γ and σ are 0.3 and 0.6 respectively. The process is repeated k times with a batch of b samples selected in each iteration and added to D_{train} .

Results of the 4 experiments are compiled and presented in Fig. 7, 8 and 9, each highlighting the performance of the 4 sampling strategies for a particular evaluation parameter in 4 datasets.

Fig. 7 presents comparative results with respect to evaluation parameter F1-score. As shown in (7), F1-score is defined as the harmonic mean of precision and recall. The measure focuses on the number of False Positive and False Negative instances detected by the model. A high F1-score implies that the model has low False Positive and low False Negative instances. The outcome is recorded for datasets D1, D2, D3 and D4. In each graph, F1-score is plotted against number of samples in D_{train} . As seen in Fig. 7, E-MMSIM outperforms ES and LC sampling and emerges as the top performing QS achieving highest F1-score of 0.89, 0.93, 0.87 and 0.79 in datasets D1, D2, D3 and D4 respectively. An average improvement of 0.77% in F1-score is achieved across all iterations compared to other 3 QS functions. It is to be noted that F1-score curve for E-MMSIM is steeper in majority of the datasets. This is indicative of E-MMSIM gaining higher F1-score at an early stage of iteration compared to other QS functions. The observation is supported by the precision-recall curve shown in Fig. 8. The plot captures the precision and recall of the classifier when it is trained with only 15% of the budget. Another important aspect to note is that ES and LC performance curves overlap each other for a significant number of iterations in all the datasets. This implies that the 2 query strategies are not mutually independent and are selecting the same set of samples for annotation in the concerned iterations. RS does not perform well, recording lowest F1-score in all the datasets. This is consistent with the paradigm that emphasizes the importance of query functions in active learning iterations over a random approach for selecting samples for manual labelling [20], [21].

In Fig. 9, the AUC achieved by the 4 QS functions is presented. E-MMSIM outperforms the other QS functions in 3 out of 4 datasets achieving highest AUC of 0.96, 0.87, 0.86 respectively. The ROC for the 4 QS functions is also shown in Fig. 10. The curve records the TPR and FPR of the classifiers at different decision thresholds when only 15% of the budget is used for training. An average improvement of

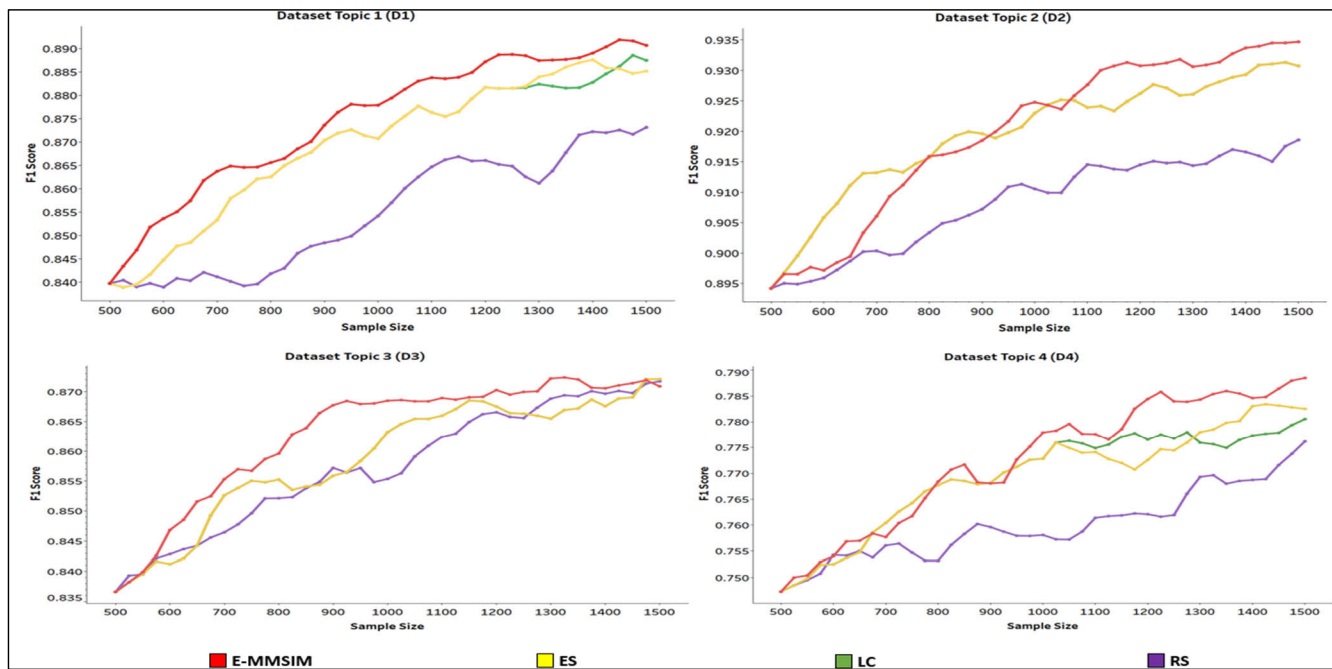


FIGURE 7. Performance of QS-F1-Score.

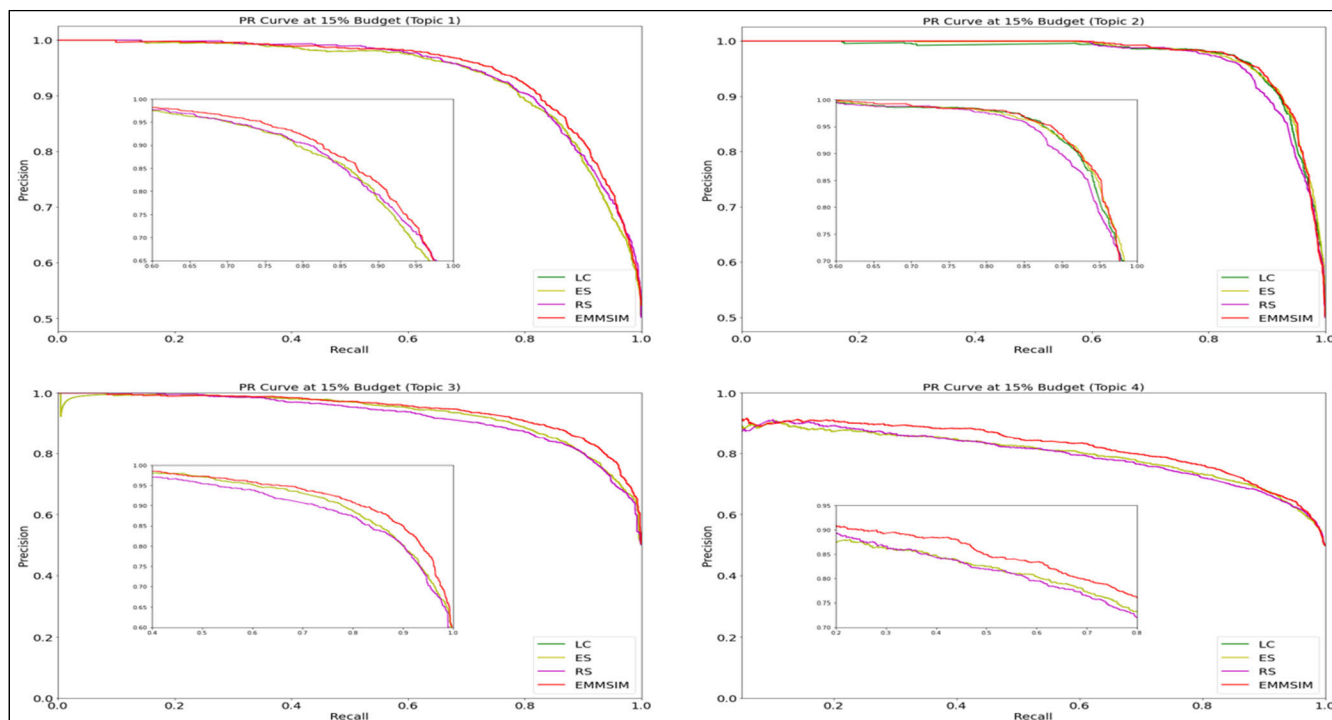


FIGURE 8. Precision-recall Curve at 15% budget.

0.51% in AUC is achieved across all iterations compared to other 3 QS functions. ES and LC strategies perform better in the second dataset achieving highest AUC of 0.98. Similar to F1-score, in AUC plots, ES and LC performance curves overlap each other for a significant number of iterations in

all the datasets implying mutual interdependence between the two QS functions. RS is the least performing strategy in all the datasets.

Fig. 11 presents comparative results with respect to evaluation parameter accuracy. As shown in (9), accuracy

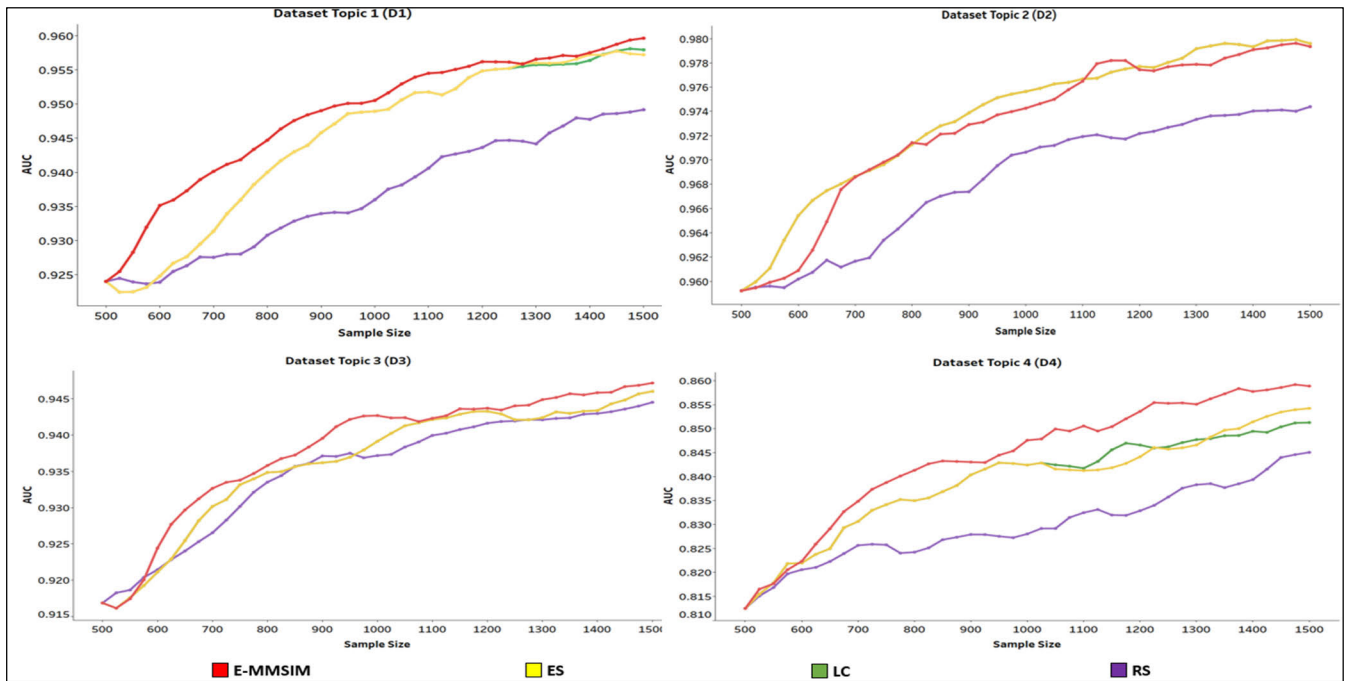


FIGURE 9. Performance of QS-AUC.

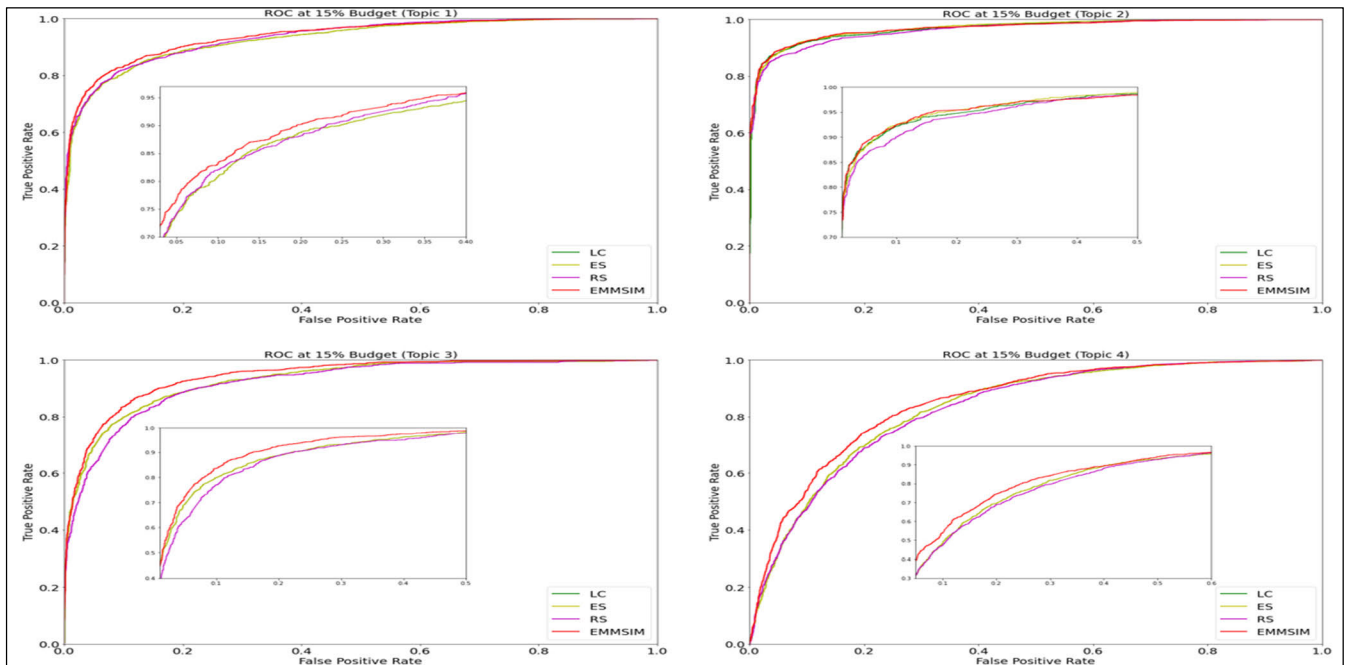


FIGURE 10. ROC at 15% budget.

focuses on the number of True Positive and True Negative instances detected by a classifier. The experiments show that the accuracy and F1-score curves are similar in the nature of outcome. E-MMSIM outperforms other QS functions in all datasets with highest accuracy of 0.89, 0.94, 0.87 and 0.78. An average improvement of 0.79% in accuracy is achieved

across all iterations compared to other 3 QS functions. Performance of E-MMSIM is indicative of the algorithm’s ability to select samples that are not redundant and are most representative and informative. This leads to state-of-the-art performance at a very early stage of active learning process. ES and LC accuracy plots overlap across significant

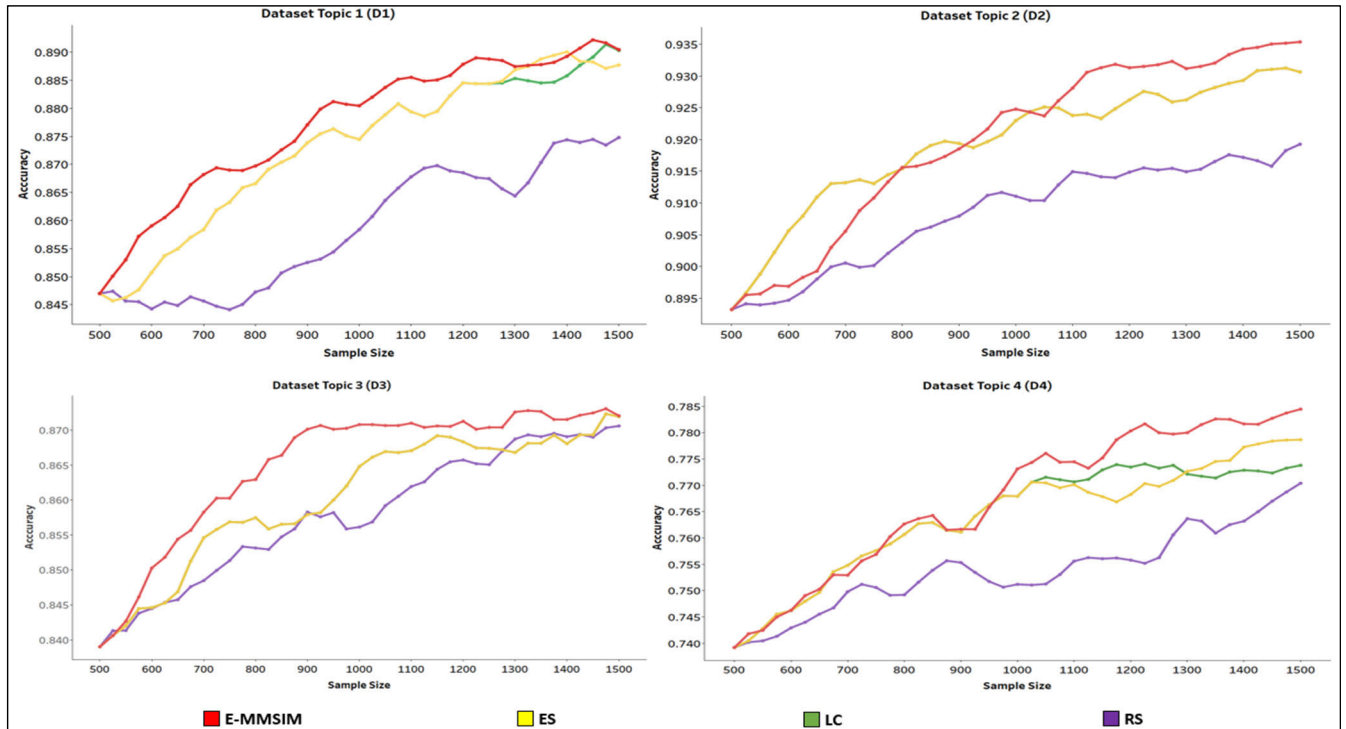


FIGURE 11. Performance of QS–Accuracy.

number of iterations. RS is the least performing strategy in all datasets.

The proposed QS strategy E-MMSIM performs better than Entropy-based and LC-based query functions for topic classification in 4 datasets for all the evaluation parameters. The algorithm selects samples for annotation using hierarchical approach, combining entropy, Hobohm-1 algorithm and misclassification similarity, to derive the representativeness and the informativeness value of a sample. Fig. 12 is a plot of average rank attained by each QS function against the percentage of budget. As seen in the figure, E-MMSIM maintains an average rank of 1 for all evaluation parameters in the 4 datasets. Additionally, it maintains the average rank across all the percentage of samples used. The experiments provide evidence that LC and ES functions are inter-dependent in nature and may not be able to provide additional benefit in terms of sample selection, particularly in the initial iterations of active learning. In Fig. 12, a divergence in rank for ES and LC occurs once more than 60% of budget is used for training. This implies that additional performance gain for ES and LC-based sampling can be observed only when larger number of samples are included in the training set. The observed inter-dependent nature of ES and LC heuristics is in agreement with findings of a few other studies [20], [21]. RS is the least performing algorithm at rank 4.

One of the practical insights of the proposed query strategy is divergence from traditional approaches that scan the entire dataset for computing similarity to derive representativeness of a sample. For example, a dataset containing n samples

will need similarity computations that are of the order of n^2 for each iteration. Using E-MMSIM, the number of computations is greatly reduced because the algorithm only evaluates a subset of the ordered dataset, and not the complete dataset, to derive sample representativeness. As a result, the computational time is expected to remain within the tolerance limits, rendering it suitable for large datasets. Second, E-MMSIM incorporates mis-classification similarity as a component to select most informative samples that are suitable for manual annotation. This approach is the main contributing factor for reducing the number of training samples to achieve state-of-the-art performance. E-MMSIM selects only the highly representative and informative samples for manual annotation. This approach reduces the high cost that is associated with annotation and renders the algorithm suitable for cases where budget is constrained.

E. DERIVING QUALITATIVE FEATURES FOR CHURN PREDICTION

As discussed before, the study uses E-MMSIM to train 4 topic classifiers with 1,500 labeled instances for each topic of churn prediction. Hence, a total of 6,000 labeled textual instances are used to train the 4 topic classifiers. Each topic classifier is utilized to further classify the set of 2,75,124 interaction messages. Hence, each message can belong to more than 1 topic of churn prediction; a case of multi-label text classification. The property is depicted by incorporating 4 binary qualitative topic variables against each interaction message of a customer. Fig. 13 shows an

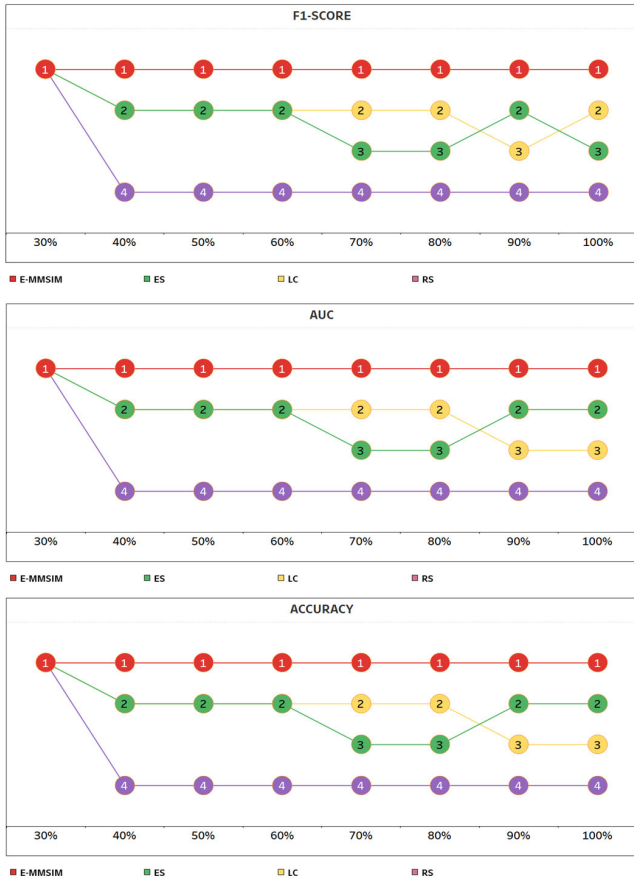


FIGURE 12. Average ranked performance of each QS.

Customer ID	Message Origin	Processed Message	Churn Topic 1	Churn Topic 2	Churn Topic 3	Churn Topic 4
Customer 1	Email	XXX	1	0	0	1
Customer 1	Email	XXX	0	0	1	1
Customer 1	Web	XXX	0	0	1	0
Customer 2	Chat	XXX	0	0	0	0
Customer 2	Email	XXX	0	0	1	1
Customer 2	Email	XXX	0	0	1	0
Customer 3	Chat	XXX	1	0	0	0
Customer 3	Chat	XXX	1	0	0	0
Customer 4	Email	XXX	0	0	1	0
Customer 4	Web	XXX	0	0	0	0

FIGURE 13. Qualitative features.

Customer ID	Customer_age	Customer_product_usage	Customer_region	∑ Churn Topic 1	∑ Churn Topic 2	∑ Churn Topic 3	∑ Churn Topic 4
Customer 1	12	4	A	1	0	2	2
Customer 2	56	2	G	0	0	2	1
Customer 3	32	5	A	2	0	0	0
Customer 4	23	1	B	0	0	1	0

FIGURE 14. Quantitative and aggregated qualitative features.

example of the dataset with qualitative features. The presence or absence of a churn-related topic in a message is denoted by 1 or 0 respectively. The qualitative features in the dataset are then aggregated and transformed for each customer to quantify the nature of interaction between the customer and the service provider over a period of time. Fig. 14 represents our structured variables that are combined with qualitative features that are numerically quantified for each customer.

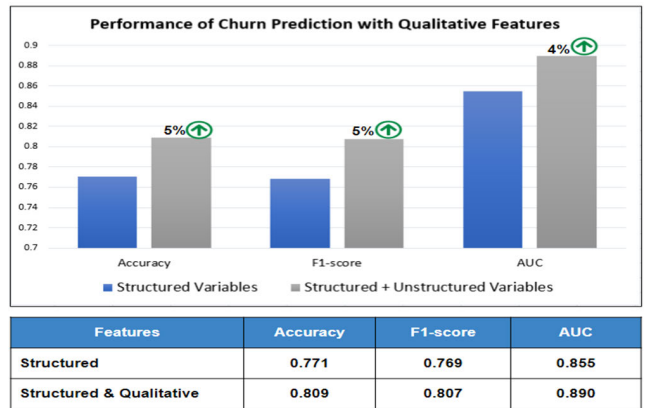


FIGURE 15. Enhanced performance with qualitative features.

F. SIGNIFICANCE OF QUALITATIVE FEATURES

Qualitative features are information-rich and when seamlessly integrated with structured variables, can strengthen a classifier's performance to a great extent. To gain evidence to support this paradigm, two experiments are conducted to predict churn, one that uses 78 structured variables alone, and another that uses 78 structured variables along with 4 qualitative attributes. Both the experiments use Random Forest classifier to predict churn. The results are mean values of 5-fold cross validation, repeated 5 times.

Results are shown in Fig. 15 and indicate enhanced predictive capability of churn classifiers by using qualitative and quantitative features. The performance is found to improve by 5%, for both accuracy and AUC evaluation parameters, and by 4% for F1-score. The study helps to Enhanced Performance with Qualitative Features emphasize the need to shift from the current model-centric approach, to a more data-centric approach in the domain of classification.

VI. CONCLUSION

The study contributes to tackle the problem of redundancy in samples selected for annotation in active learning by means of a new hierarchical algorithm called E-MMSIM. The algorithm quantifies suitability of a sample for annotation by combining informativeness and representativeness as a measure. Additionally, the heuristic incorporates misclassification similarity to rank the samples for annotation. This is in stark contrast to existing approaches that need pre-defined hyperparameters and do not necessarily focus on eliminating redundancy in samples. Besides, informative value of misclassified samples is rarely considered in experiments. E-MMSIM outperforms popular query functions for topic classification in all the 4 datasets for majority of the evaluation parameters. Additionally, the study finds that performance curves of ES and LC query functions overlap each other for a significant number of iterations of active learning workflow. This is indicative that ES and LC strategies are highly inter-dependent and result in selecting the same set of samples for annotation.

E-MMSIM helps to derive qualitative features using only 2% of the total samples for training. These features when integrated with quantitative features of a churn dataset, led to a 5% gain in both accuracy and F1-score, and 4% gain in AUC. The findings emphasize the need to shift from a model-centric approach, to a data-centric approach in the context of machine learning classification. This is the second contribution of the study.

There are few limitations that should be highlighted. Firstly, the inclusion of misclassification similarity may lead to increased bias of the learning classifier. Although this drawback can be overcome by making periodic and continuous updates to the test set, more experiments are needed to study the impact of this change, and this approach itself could be a direction for future research. Secondly, the condition of informativeness and representativeness measure is stringent in nature and may lead to an indefinite iteration. Future research could focus on refining the measure of informativeness-representativeness by introducing multiple logical operators that are less stringent. Thirdly, the study evaluates E-MMSIM for the use case of binary topic classification. Future experiments that utilize E-MMSIM on datasets of varied nature, will help to establish the effectiveness of the proposed QS function.

REFERENCES

- [1] A. D. Sappa and J. Vitrià, *Multimodal Interaction in Image and Video Applications*. Berlin, Germany: Springer, 2013, doi: [10.1007/978-3-642-35932-3](https://doi.org/10.1007/978-3-642-35932-3).
- [2] B. Settles, "Active learning," *Synth. Lectures Artif. Intell. Mach. Learn.*, vol. 6, no. 1, pp. 1–114, Jun. 2012, doi: [10.2200/S00429ED1V01Y201207AIM018](https://doi.org/10.2200/S00429ED1V01Y201207AIM018).
- [3] S. Budd, E. C. Robinson, and B. Kainz, "A survey on active learning and human-in-the-loop deep learning for medical image analysis," *Med. Image Anal.*, vol. 71, Jul. 2021, Art. no. 102062, doi: [10.1016/j.media.2021.102062](https://doi.org/10.1016/j.media.2021.102062).
- [4] X. Wang and J. Zhai, *Learning With Uncertainty*. Boca Raton, FL, USA: CRC Press, 2016, doi: [10.1201/9781315370699](https://doi.org/10.1201/9781315370699).
- [5] D. Tuia, F. Ratle, F. Pacifici, M. F. Kanevski, and W. J. Emery, "Active learning methods for remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 7, pp. 2218–2232, Jul. 2009, doi: [10.1109/TGRS.2008.2010404](https://doi.org/10.1109/TGRS.2008.2010404).
- [6] J. Zhu, H. Wang, B. K. Tsou, and M. Ma, "Active learning with sampling by uncertainty and density for data annotations," *IEEE Trans. Audio, Speech, Language Process.*, vol. 18, no. 6, pp. 1323–1331, Aug. 2010, doi: [10.1109/TASL.2009.2033421](https://doi.org/10.1109/TASL.2009.2033421).
- [7] D. Wang and Y. Shang, "A new active labeling method for deep learning," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2014, pp. 112–119, doi: [10.1109/IJCNN.2014.6889457](https://doi.org/10.1109/IJCNN.2014.6889457).
- [8] Z. Guochen, "Four uncertain sampling methods are superior to random sampling method in classification," in *Proc. 2nd Int. Conf. Artif. Intell. Educ. (ICAIE)*, Jun. 2021, pp. 209–212, doi: [10.1109/ICAIE53562.2021.00051](https://doi.org/10.1109/ICAIE53562.2021.00051).
- [9] R. B. C. Prudencio, C. Soares, and T. B. Ludermir, "Uncertainty sampling methods for selecting datasets in active meta-learning," in *Proc. Int. Joint Conf. Neural Netw.*, Jul. 2011, pp. 1082–1089, doi: [10.1109/IJCNN.2011.6033343](https://doi.org/10.1109/IJCNN.2011.6033343).
- [10] K. Wang, D. Zhang, Y. Li, R. Zhang, and L. Lin, "Cost-effective active learning for deep image classification," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 12, pp. 2591–2600, Dec. 2017, doi: [10.1109/TCSVT.2016.2589879](https://doi.org/10.1109/TCSVT.2016.2589879).
- [11] S. Wen, T. M. Kurc, L. Hou, J. H. Saltz, R. R. Gupta, R. Batiste, T. Zhao, V. Nguyen, D. Samaras, and W. Zhu, "Comparison of different classifiers with active learning to support quality control in nucleus segmentation in pathology images," *AMIA Summits Transl. Sci. Proc.*, to be published.
- [12] W. H. Beluch, T. Genewein, A. Nurnberger, and J. M. Kohler, "The power of ensembles for active learning in image classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 9368–9377.
- [13] K. Konyushkova, R. Sznitman, and P. Fua, "Geometry in active learning for binary and multi-class image segmentation," *Comput. Vis. Image Understand.*, vol. 182, pp. 1–16, May 2019, doi: [10.1016/j.cviu.2019.01.007](https://doi.org/10.1016/j.cviu.2019.01.007).
- [14] H. U. Hobohm, M. Scharf, R. Schneider, and C. Sander, *Selection Representative Protein Data Sets*. Cambridge, U.K.: Cambridge Univ. Press, 1992.
- [15] T. B. Alakus and I. Turkoglu, "Comparison of deep learning approaches to predict COVID-19 infection," *Chaos, Solitons Fractals*, vol. 140, Nov. 2020, Art. no. 110120, doi: [10.1016/j.chaos.2020.110120](https://doi.org/10.1016/j.chaos.2020.110120).
- [16] N. Q. K. Le, Q.-T. Ho, V.-N. Nguyen, and J.-S. Chang, "BERT-promoter: An improved sequence-based predictor of DNA promoter using BERT pre-trained model and SHAP feature selection," *Comput. Biol. Chem.*, vol. 99, Aug. 2022, Art. no. 107732, doi: [10.1016/j.compbiolchem.2022.107732](https://doi.org/10.1016/j.compbiolchem.2022.107732).
- [17] N.-Q.-K. Le and B. P. Nguyen, "Prediction of FMN binding sites in electron transport chains based on 2-D CNN and PSSM profiles," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 18, no. 6, pp. 2189–2197, Nov. 2021, doi: [10.1109/TCSB.2019.2932416](https://doi.org/10.1109/TCSB.2019.2932416).
- [18] J. N. Mandrekar, "Receiver operating characteristic curve in diagnostic test assessment," *J. Thoracic Oncol.*, vol. 5, no. 9, pp. 1315–1316, Sep. 2010, doi: [10.1097/JTO.0b013e3181ec173d](https://doi.org/10.1097/JTO.0b013e3181ec173d).
- [19] L. Wynants, "Prediction models for diagnosis and prognosis of COVID-19: Systematic review and critical appraisal," *Brit. Med. J.*, vol. 369, Apr. 2020, Art. no. m1328, doi: [10.1136/bmj.m1328](https://doi.org/10.1136/bmj.m1328).
- [20] J. S. Smith, B. Nebgen, N. Lubbers, O. Isayev, and A. E. Roitberg, "Less is more: Sampling chemical space with active learning," *J. Chem. Phys.*, vol. 148, no. 24, Jun. 2018, Art. no. 241733, doi: [10.1063/1.5023802](https://doi.org/10.1063/1.5023802).
- [21] Y. Yang and M. Loog, "A benchmark and comparison of active learning for logistic regression," *Pattern Recognit.*, vol. 83, pp. 401–415, Nov. 2018, doi: [10.1016/j.patcog.2018.06.004](https://doi.org/10.1016/j.patcog.2018.06.004).



SOUMI DE (Member, IEEE) received the B.Sc. degree (Hons.) in physics from Calcutta University, West Bengal, India, in 2000, and the M.C.A. degree in computer applications from Visva-Bharati University, West Bengal, in 2003. She is currently pursuing the Ph.D. degree in data science with CHRIST (Deemed to be University), Karnataka, India. Since 2004, she has been working with several multi-national corporations in the domain of business intelligence and analytics. Her research interests include decision science, natural language processing, and predictive analytics. She has received the Top Performer Award, in 2015, for her contribution in automating analytical solutions in a multi-national company.



P. PRABU received the Ph.D. degree in computer applications from Anna University, India. He is currently working as an Assistant Professor with the Department of Computer Science, CHRIST (Deemed to be University), Karnataka, India. He has more than 14 years of experience in teaching and industry. His research interests include software engineering, web service, deep learning, the IoT, and mobile application.