An active learning pipeline to automatically identify candidate terms for a CDSS ontology—measures, experiments, and performance

Shailesh Alluri, MS[1], Keerthana Komatineni*, MS[1], Rohan Goli*, MS[1], Richard D. Boyce, PhD[2], Nina Hubig, PhD[1], Hua Min, PhD[3], Yang Gong, PhD[4], Dean F. Sittig, PhD[4], David Robinson, MD[5], Paul Biondich, MD[6], Adam Wright, PhD[7], Christian Nøhr, PhD[8], Timothy Law, DO[9], Arild Faxvaag, PhD[10], Ronald Gimbel, PhD[11], Lior Rennert, PhD[11], Xia Jing[#], PhD[11]

1School of Computing, College of Engineering, Computing and Applied Sciences, Clemson University, Clemson, SC, USA; 2Department of Biomedical Informatics, School of Medicine, University of Pittsburgh, Pittsburgh, PA, USA; 3Department of Health Administration and Policy, College of Public Health, George Mason University, Fairfax, VA, USA; 4Department of Clinical and Health Informatics, McWilliams School of Biomedical Informatics, The University of Texas Health Sciences Center at Houston, Houston, TX, USA; 5Independent Consultant, Cumbria, UK; 6Department of Pediatrics, School of Medicine, Indiana University School of Medicine, Indianapolis, IN, USA; 7Department of Medicine, School of Medicine, Vanderbilt University Medical Center, Nashville, TN, USA; 8Department of Sustainability and Planning, The Technical Faculty of IT and Design, Danish Center for Health Informatics, Aalborg University, Aalborg, Denmark; 9Ohio Musculoskeletal and Neurologic Institute, Ohio University, Athens, OH, USA; 10Department of Neuromedicine and Movement Science, School of Medicine, Norwegian University of Science and Technology, Trondheim, Norway; 11Department of Public Health Sciences, College of Behavioral, Social and Health Sciences, Clemson University, Clemson, SC; *Coauthors contributed equally; #Corresponding author.

ABSTRACT:

In this paper, we present an active learning framework designed to automatically identify key phrases relevant to a Clinical Decision Support System (CDSS) ontology. CDSS ontology can play a critical role in standardizing medical vocabulary and enabling seamless data integration across healthcare systems. Traditional methods for ontology development are manual, labor-intensive, and require significant domain expertise. Our approach combines a BiLSTM-CRF model with humans in the loop active learning pipeline to progressively incorporate human experts' feedback to improve its accuracy. We implement uncertainty sampling as our core document selection strategy, prioritizing instances where the model exhibits low confidence for human review. We introduce new uncertainty aggregation methods—KPSum, KPAvg, DOCSum, and DOCAvg—which, in combination with uncertainty measures such as Maximum Token Probability (MTP), Token Entropy (TE), and Margin, are used to calculate document-level confidence scores. These methods improve the document selection process in Active Learning and make the document selection process transparent and replicable, ensuring that the most informative documents are prioritized for annotation. This study underscores the value of active learning in facilitating ontology development, which can play a significant role in reducing

manual effort and facilitating human experts, especially during the long term maintenance stage of ontology.

INTRODUCTION:

Clinical Decision Support Systems (CDSS) have become a cornerstone in modern healthcare, aiding medical practitioners by offering data-driven recommendations to improve patient care [Ref 1]. Ontology is an enabling technology of the Semantic Web technologies and plays a critical role in information interoperability [Ref 2]. A biomedical ontology, defines the domain entities and their relationships, can be used in the domain (for example CDSS) in multiple ways, such as to develop reusable CDSS rules. The ontologies along with terminologies are crucial in standardizing clinical vocabulary and ensuring interoperability across different healthcare systems [Ref 3]. However, building and maintaining ontologies typically is a manual effort and can be a challenging task, as it is labor intensive and it requires domain expertise. In the long term, the ontology requires continuous updates to include newly discovered medical terms, procedures, and treatments. As medical knowledge rapidly evolves, the task of maintaining ontologies manually becomes increasingly challenging. Without consistent maintenance after construction, the outdated ontologies could lead to gaps in outdated domain information. Although description logics [Ref 4] and dedicated teams can overcome some of the challenges, automating some of the steps can provide significant relief in long term ontology maintenance.

This paper addresses these limitations by proposing an active learning-based pipeline that automates the identification of key phrases from unstructured CDSS publications. The key phrases can include a single word, nouns or verbs, or a phrase. The key phrases can be used as candidates to be included in the CDSS ontology. Our approach leverages deep learning models to systematically select and highlight important terms.The model then is refined through iterative human feedback. This reduces the reliance on solely manual selection while ensuring the ontology remains accurate and current.The approach is particularly important in long term CDSS ontology maintenance. By incorporating feedback from human domain experts (HDEs), the model continuously improves its performance, refining its predictions of context-specific key phrases and evolving, as the new HDE labeled data become available. Our pipeline prioritizes instances where the model is uncertain or where consensus is mostly needed, enabling targeted human review to resolve ambiguities effectively.

The key contributions of this work are as follows:

1. **Noval aggregation methods for calculating confidence scores in active learning**: We propose new document-level confidence aggregation methods—KPSum, KPAvg, DOCSum, and DOCAvg—that prioritize evidence for HDE review more effectively. These methods leverage established uncertainty sampling strategies, such as Maximum Token Probability (MTP), Margin, and Token Entropy [Ref 5], to improve the document

selection process in Active Learning, ensuring that the most informative documents are chosen for HDE annotation.

2. **Seamless integration of an active learning pipeline**:
   This work introduces a novel active learning framework for key phrase identification with a continuous HDE feedback loop. This approach reduces the pure HDE manual annotations while iteratively refining the active learning model to adapt to evolving data distributions over time.
3. **Systematic comparison of various strategies:**
   We conduct a systematic evaluation of multiple Active Learning strategies(e.g. MTP DocAvg Actual, MTP DocAvg Reverse, MTP KPSum Actual, TE KPSum Actual)**.** We also compare these strategies against a Random document selection, which was used as the baseline, to assess the advantages of uncertainty-based document selection. By analyzing their impact on F1, recall, and weighted F1 values over successive Active Learning cycles, we provide robust evidence and insights into comparing the effectiveness of each strategy in improving model performance and annotation efficiency.
4. **Improvement in key phrase identification**:
   Through iterative Active Learning cycles, our approach achieves faster and more accurate identification of key phrases, reducing the dependence on manual annotation.

The remainder of this paper is structured as follows**:** In Section 2, we review related work on key phrase identification and active learning approaches. Section 3 (Methodology) describes the integration of human feedback and continuous learning mechanisms. It also presents our proposed methodology, including a brief introduction of the deep learning model architecture with an active learning loop, and our novel confidence score calculation methods using our proposed uncertainty aggregation methods (KPSum, KPAvg, DOCSum, and DOCAvg). Section 4 (Experiments) outlines the experimental setup and presents our results, while Section 5 interprets the primary experiment results with brief discussion and analysis and future work. Section 6 (Conclusion) concludes with the implications of our findings.

Section 2: Literature

Identifying key phrases from textual data is a critical task in natural language processing (NLP), forming the foundation for downstream applications such as ontology development, information retrieval, and text summarization. Traditional methods, including rule-based and statistical approaches like Term Frequency-Inverse Document Frequency and Rapid Automatic Keyword Extraction [Ref 6], have been widely adopted for key phrase extraction. However, these methods often fail to generalize across domains, prompting the adoption of machine learning-based models, such as conditional random fields (CRFs) [Ref 7] and sequence labeling models, for named entity recognition (NER) and key phrase identification.

The advent of deep learning has revolutionized the NER landscape. Models such as Bidirectional Long Short-Term Memory Network - Conditional Random Fields (BiLSTM-CRF) [Ref 8] and transformer-based architectures like Bidirectional Encoder Representations from Transformers (BERT) [Ref 9] have set new benchmarks in NER tasks. Notably, the work by

Alzaidy et al. utilized [Ref 10] a BiLSTM-CRF model for key phrase identification, demonstrating its ability to effectively leverage hierarchical context and transfer learning techniques for domain-specific applications. Our work builds on this foundation, extending it with an active learning framework to reduce annotation costs while maintaining high accuracy in key phrase identification.

Active learning (AL) is a subfield of machine learning that aims to achieve high model performance with minimal labeled data by strategically selecting the most informative samples for annotation [Ref 11]. Active learning has proven to be an effective strategy to enhance performance in tasks with limited labeled data, such as NER. Settles (2009) [Ref 12] provides a comprehensive overview of active learning strategies, highlighting uncertainty sampling, query-by-committee, and expected error reduction as key techniques. Shen et al. (2017) [Ref 13] employed uncertainty sampling with deep learning models, effectively reducing the labeling effort required for text classification. Similarly, Siddhant and Lipton (2018) [Ref 14] explored active learning in low-resource NER tasks, demonstrating the benefits of selectively annotating high-value samples. Caufield, et al also explored using large language models (GPT-3.5 and GPT-4) with structured prompts to populate ontologies, to leverage the most advanced technology to achieve similar goals [Ref 15].

Integrating active learning into key phrase identification has gained attention in recent years. For instance, Radmard et al. (2021) [Ref 16] utilized active learning to iteratively refine Named Entity Recognition (NER) models, focusing on identifying ambiguous or novel terms in unstructured text. Margatina et al. (2021) [Ref 17] extended active learning techniques to transformer models like BERT, achieving significant performance gains with reduced annotation costs. These studies underscore the potential of active learning in improving the efficiency and accuracy of NER systems.

Our work differs from above-mentioned work in two key ways. First, unlike prior studies that often rely on a small manually labeled seed set, our approach begins with a synthetically labeled training set, requiring the model to adapt to real-world patterns progressively. Second, and more importantly, we introduce novel uncertainty aggregation strategies designed specifically for document-level sampling. Traditional token-level uncertainty scores are inadequate in our setting where the sampling unit is an entire document. To address this, we propose several methods—including KPSUM, KPAVG, DOCSUM, and DOCAVG—to aggregate uncertainty from tokens or keyphrases at the document level. These strategies enable more effective document selection and represent a significant advancement in adapting active learning to keyphrase identification at scale with transparency.

Our work contributes to this growing body of research by introducing an active learning framework specifically designed to enhance the accuracy of key phrase identification in the CDSS domain. By leveraging uncertainty sampling and human-in-the-loop feedback mechanisms, our approach enables the iterative refinement of our active learning pipeline, achieving more accurate and scalable key phrase identification. This framework ultimately supports the broader goal of ontology development by providing high-quality, domain-specific key phrases for HDE to review.

Section 3: METHODOLOGY:

3.1 Active Learning Overview

Active learning (AL) aims to achieve high model performance with minimal labeled data by strategically selecting the most informative samples for annotation [Ref 18]. In a typical supervised learning setting, a large amount of labeled data is required to train a model effectively. However, in many real-world applications, such as NER or key phrase identification, obtaining labeled data is expensive and time-consuming due to the need for domain expertise. Active learning addresses this challenge by iteratively selecting the data points that the model finds most uncertain or informative and presenting them for human annotation. This process reduces the overall annotation effort while maximizing the model's learning efficiency.

An Active Learning Cycle refers to the learning process between two stages, e.g., θ_{x} to θ_{x+1}, where θ_{x} represents the model state after x cycles of Active Learning. Each cycle involves selecting the most informative samples from the data pool, obtaining human annotations, and retraining the model with the newly human labeled data (Figure 1). This iterative approach ensures that the model continuously improves by focusing on the most uncertain or informative samples in each cycle. The model will obtain incrementally improving performance through each cycle.

Our active learning cycle operates as follows:

At stage θ_{x}, a BiLSTM-CRF model is used to classify tokens into five possible tags: B-KP (Beginning of a key phrase), I-KP (Inside a key phrase), O-KP (Outside a key phrase), Start (Start of the sequence), and Stop (End of the sequence). The model assigns probabilities to each token, indicating the likelihood of belonging to one of these categories. These probabilities serve as the foundation for uncertainty measures, which are used to compute confidence scores for each document in the Active Learning data pool.

Once the confidence scores are calculated (sections 3.3.1 and 3.3.2 will introduce the detailed methods about calculations), the documents in the Active Learning data pool are sorted based on these scores. The sorting ensures that the most informative documents—those that the model is most uncertain about—are prioritized for HDE annotation. The order of selection (explained in detail later in section 4.1.1) determines how these documents are chosen. A subset of documents is then selected and sent to HDEs for annotation.

After annotation, the HDE labeled documents are added to the training data and removed from the Active Learning data pool to prevent multiple copies of the same document being labeled again. This step is essential to avoid overfitting, as repeatedly training on the same document could reinforce biases rather than improving the model's generalizability.

Once the newly labeled documents are incorporated into the training dataset, the BiLSTM-CRF model is retrained using the updated training dataset. This newly trained model, denoted as

θ□□□, then undergoes the same Active Learning selection process in the next cycle. This iterative process continues until a stopping criterion (such as performance saturation) is met. By systematically incorporating human-labeled data and prioritizing informative samples, this Active Learning cycle enhances the model's ability to classify key phrases more effectively with each cycle.

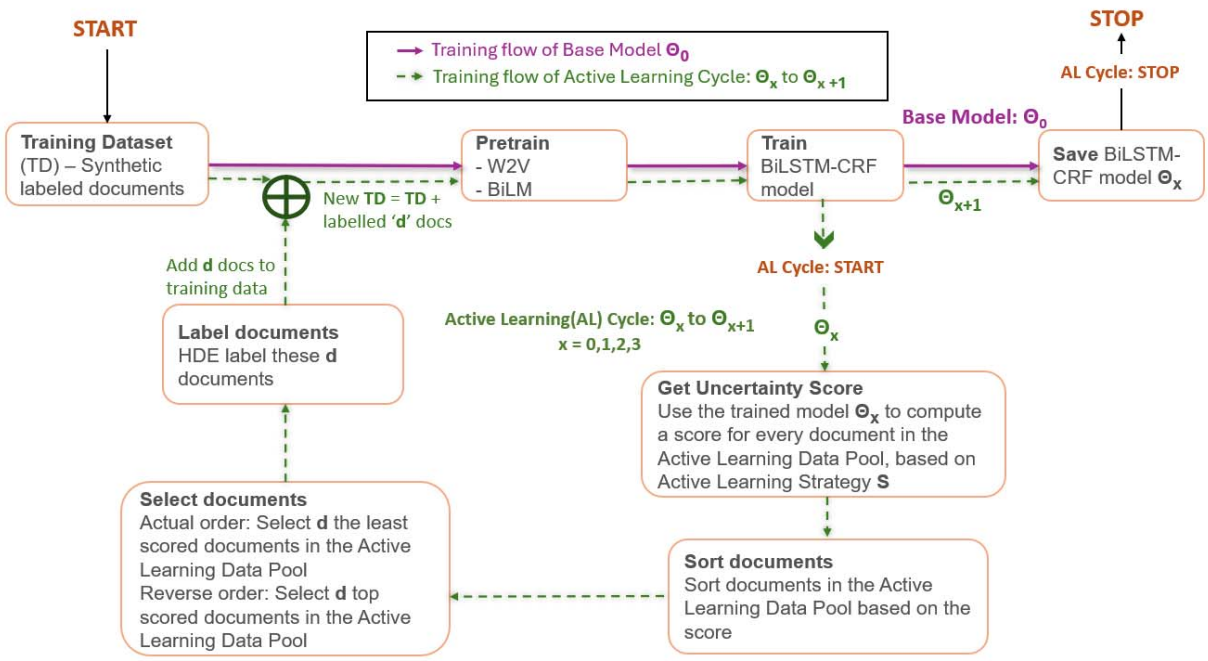

Figure 1: Active Learning Cycle for automatic key phrase identification

3.3 Active Learning Framework

Our framework begins with a BiLSTM-CRF model, which serves as the foundational model (denoted as θ□) for key phrase identification. Initially, θ□ is trained on 500 synthetically-labeled CDSS documents and our prior publication detailed the process [Ref 10]. Following this, the framework employs a pool-based Active Learning (AL) strategy, as outlined by Settles et al. [Ref 12], to iteratively enhance the model by selecting and labeling the most informative documents from the unlabeled documents pool. This process allows the model to progressively improve its performance by leveraging uncertainty sampling techniques, focusing on the most uncertain data points to maximize learning efficiency.

Figure 2 illustrates the entire lifecycle of our Active Learning experiment, showing the iterative model learning and improvement process from θ□ to θ□. Each transition represents an Active Learning cycle, where new human-reviewed documents are incorporated into the training dataset.

- θ□: The initial model (θ□) is trained solely on 500 synthetically labeled documents, as described in Goli R et al [Ref 10].
- θ□: After the first Active Learning cycle, the model is retrained on 500 synthetic documents + 5 selected documents reviewed by HDE.
- θ□: The training dataset expands further, now including 500 synthetic documents + 10 HDE-reviewed documents (Previous 5 from θ□ + 5 new documents).
- θ□: The model is retrained on 500 synthetic documents + 15 HDE-reviewed documents (previous 10 from θ□ and θ□ + 5 new documents).
- θ□: The final cycle includes 500 synthetic documents + 20 HDE-reviewed documents (previous 15 from θ□, θ□, and θ□ + 5 new documents), continuing the refinement process.

As the model progresses through θ□ to θ□, it continuously learns from the most informative samples, improving its key phrase identification accuracy.

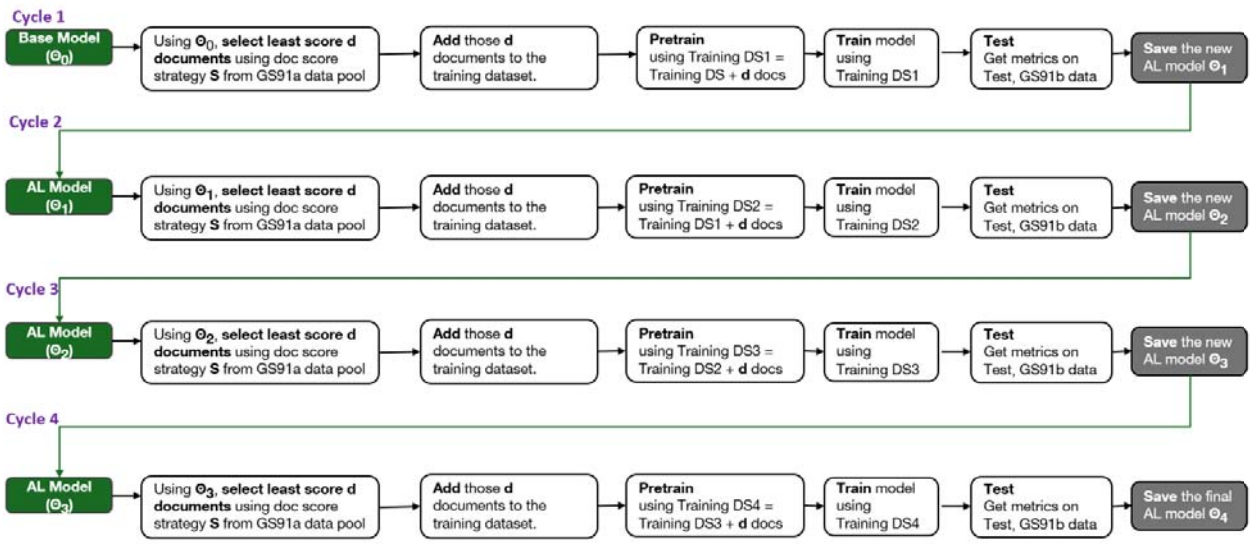

Figure 2: Active Learning life cycle examples from θ□ to θ□ with elaborated steps

Our active learning framework incorporates both established and newly developed methodologies.

3.3.1 Uncertainty Sampling Strategies: To identify sample documents for HDE annotation, we utilize well-established uncertainty metrics such as Minimum Token Probability (MTP), Margin, and Token Entropy. These metrics help identify tokens or sequences for which the model's predictions exhibit the lowest confidence.

Minimum Token Probability (MTP) [Ref 5]

The MTP assesses uncertainty by considering the highest probability assigned to any label for a token. A lower maximum probability indicates greater uncertainty, as the model lacks confidence in any single prediction.

$$\phi^{MTP}(x) = 1 - \min_i \max_j P(y_i = j \mid x_i; A)$$

Where j ranges over all possible token tags. A is a parameter called the transition matrix. X_i is the i□th token in the sentence, and y_i is the tag that the i□th token can assume.

Token Entropy [Ref 19]

The **Token Entropy** method measures uncertainty by calculating the entropy of the probability distribution across all possible tags for each token. Greater entropy indicates less information available to the model to make a confident prediction for a given token. By selecting tokens with the greatest entropy for AL, the system becomes efficient at gaining information for future classification tasks. By selecting tokens (or documents containing tokens) with the highest entropy, this method effectively identifies the most ambiguous documents for HDE review.
Token Entropy is particularly well-suited for key phrase identification tasks, as it provides a detailed, probability-based view of uncertainty, allowing for precise targeting of areas requiring clarification.

$$\phi_{TE} = -\sum_{j=1}^{M} P(y_i = j \mid x_i; A) \log P(y_i = j \mid x_i; A)$$

Where j ranges over all possible token tags. A is a parameter called a transfer matrix. X_i is the token. Y_i is the tag that the ith token in the sentence can assume M is the number of tags, in our case it is 5, since we have 5 tags.

Margin Method [Ref 20]

The **Margin Method** calculates uncertainty by examining the difference between the probabilities of the two most likely predicted labels for a token. A smaller margin reflects greater uncertainty, as the model finds it harder to differentiate between the top predictions.
While the method still involves computing the full probability distribution, it only

utilizes the top two probabilities, making it computationally more efficient than methods like Token Entropy, which require contributions from the entire distribution. The Margin Method is particularly effective for identifying instances near classification boundaries, where the model's confidence is weakest.

$$\phi_M(x) = -\left(P(y_1^* \mid x; A) - P(y_2^* \mid x; A)\right)$$

Where A is a parameter called a transfer matrix. and are the first most and second most likely tags for the token x respectively.

3.3.2 Document-Level Aggregation: We introduce novel methods—KPSum, KPAvg, DOCSum, and DOCAvg—to aggregate token-level confidence scores into document-level confidence scores . Gupta used similar measures, ChowSum and ChowAverage [Ref 21], both of which aggregate uncertainty at the sentence level, our approach differs in two key ways. First, we perform document-level aggregation, which better aligns with our Active Learning setup where entire documents are labeled. Second, we propose keyphrase-focused strategies—KPSum and KPAvg—that aggregate uncertainty only over tokens likely to belong to keyphrases, enabling more targeted and efficient sampling. This keyphrase-centric perspective is absent in prior work, distinguishing our method to the goals of keyphrase identification. As a result, documents can be prioritized more effectively for HDE review, enhancing the overall efficiency of the annotation process.

DocSum: This method involves summing the model uncertainty (measured by methods such as Token Entropy or Margin, explained above) across all tokens in a document. By considering the cumulative uncertainty of all tokens, **DocSum** identifies documents with the highest overall uncertainty for HDE review. However, **DocSum** tends to favor longer documents, as they naturally contain more tokens, which can lead to a bias toward selecting those documents.

DocAvg: Instead of summing the uncertainty, **DocAvg** calculates the arithmetic mean uncertainty per token in the document. This method ensures that documents with high mean uncertainty (rather than just long documents with more tokens) are selected, balancing document length and uncertainty.

KPSum (Key Phrase Sum): Similar to DocSum, this method sums the uncertainty scores of all identified key phrases (instead of all tokens) within a document. It prioritizes documents based on the collective uncertainty of the predicted key phrases, making it suitable when key phrases are the focus rather than all tokens in the document. However, KPSum can be biased toward documents with more key phrases, as it sums the uncertainty associated with

each predicted key phrase, so longer lists of predicted key phrases will contribute more to the score.

KPAvg (Key Phrase Avg): This strategy takes the arithmetic average uncertainty of the predicted key phrases per document, regardless of how many predicted key phrases are in a document. It is particularly effective for documents where only a few key phrases contribute to the overall uncertainty, ensuring that the model focuses on those key phrases that are most uncertain.

Both KPSum and KPAvg focus on the most ambiguous key phrases predicted by the BiLSTM-CRF model. These methods prioritize key phrase tokens over all tokens, unlike DOCSum and DOCAvg, which treat all tokens equally. As a result, KPSum and KPAvg ensure that both the model and HDE attention are directed toward documents containing the most uncertain key phrases.

Document Sorting and Selection: After computing the document-level confidence scores, all documents in the Active Learning data pool are sorted in descending order of their uncertainty (i.e., documents with the highest uncertainty are prioritized), which are then passed on to HDE for review.

Human Domain Expert Review: The most uncertain documents, as identified through the confidence scores, are reviewed and annotated by the HDE. The HDE review process ensures that the selected documents are correctly labeled and that any key phrases missed by the model are corrected.

Retraining the Model: Once the documents are labeled by HDE, they will be added to the training dataset and removed from the Active Learning data pool. The BiLSTM-CRF model is then retrained using the updated training dataset, which now includes the newly HDE labeled documents. This process allows the model to learn from the continuing updated correct labels and gradually improve its key phrase identification performance.

3.3.3 Comparison of Uncertainty Measures

The main distinction among the three uncertainty measures lies in their focus and computational efficiency. Token Entropy analyzes the entire probability distribution, providing a detailed measure of uncertainty suited for complex, multi-label tasks. In contrast, the Margin Method focuses on the difference between the top two predicted labels, offering a computationally efficient solution for large-scale tasks where quick AL cycles are essential. MTP, on the other hand, simplifies uncertainty assessment by considering only the highest predicted probability, making it useful for flagging low-confidence predictions in simpler tasks while being less computationally intensive than Token Entropy. Table 1 presents side by side comparison of the three measures.

Table 1. Comparison among uncertainty measures used in the active learning pipeline.

Uncertainty Measure	**Focus**	**Computational Efficiency**	**Best Suited For**
Token Entropy	Analyzes the entire probability distribution	Computationally intensive	Complex, multi-label tasks where full range of potential outcomes is critical
Margin Method	Focuses on the difference between the top two predicted labels	Computationally efficient	Large-scale tasks requiring quick cycles
Minimum Token Probability (MTP)	Considers only the highest predicted probability	Less computationally intensive than Token Entropy, and Margin Method	Identifying low-confidence predictions in simpler tasks or high-throughput scenarios.

4. Experiments

This section outlines the experimental design used to evaluate the effectiveness of various Active Learning strategies for key phrase identification. It includes the datasets used, the metrics selected for evaluation, and the document selection order strategies applied across Active Learning cycles. Table 2 summarizes the systematic evaluation of different measures and strategies and their combinations used during the experiments. The results and performance comparisons are presented in section 4.5.

4.1 Experiment preparation

4.1.1 Document Order in Active Learning Cycles

The order in which documents are selected during active learning cycles plays a significant role in shaping the model learning trajectory. We evaluate two strategies:

1. **Actual Order**: Documents are prioritized based on their inherent uncertainty (e.g., calculated by Token Entropy, Margin, or Maximum Token Probability). This strategy mimics a real-world scenario where the most uncertain documents are presented first for HDE annotation.
2. **Reverse Order**: Documents are presented in reverse order of their uncertainty. This serves as a control to evaluate whether prioritizing high-uncertainty documents truly enhances model performance compared to random or low-priority sampling.

The "Actual Order" strategy focuses on presenting the most uncertain documents first for HDE annotation, under the assumption that these documents will provide the greatest benefit in improving model performance. During the Actual Order cycles,the documents on the top with the least confident scores, so that adding them all at once can pose challenges for the model. In contrast, by prioritizing documents in Reverse Order—where the most certain documents are presented first—the selection process begins with samples that are more similar to the existing data. This approach allows the model to adjust more gradually. This consideration is particularly relevant considering our training data is synthetically generated from a small pool of documents. These comparisons allow us to assess the impact of document order on annotation efficiency and overall model performance.

4.1.2 Experiment Strategies: In each experiment, we test a specific strategy, which is a combination of the uncertainty measure(e.g., Margin, MTP, Token Entropy), document aggregator (e.g., DOCSum, DOCAvg, KPSum, KPAvg), and document order (e.g., Actual, Reverse). These combinations collectively determine how documents are prioritized for HDE annotation during Active Learning cycles. Table 2 presents the evaluation settings we implemented in the experiments. We aim to compare the performance of different Active Learning strategies in terms of recall and F1 improvement and to identify the most effective one for improving the model's performance.

Table 2. All the active learning strategies (i.e., combinations of uncertainty measure and strategy) experimented in this study

Active Learning Strategies Evaluated in the Experiments				
Uncertainty Measure/ Document Aggregator	**DOCSum**	**DOCAvg**	**KPSum**	**KPAvg**
MTP	A:NA R: NA	A: S6 R: S5	A: S8 R: S7	A: S10 R: S9
MARGIN	A:NA R: NA	A: S2 R: S1	A: S3 R: S4	A:NA R: NA
TOKEN ENTROPY	A:NA R: NA	A: S12 R: S11	A: S14 R: S13	A:NA R: NA

NA: Not Available; A: Actual order of uncertainty scores; R: Reverse order of uncertainty scores; Sx : Strategy x.

The DOCSum-based strategies are not included in the final performance comparison, as they consistently favored longer documents, potentially biasing the evaluation. However, we leveraged DOCSum during the development phase to assist in selecting hyperparameters, given its tendency to surface high-information documents early in the process.

.

4.1.3 Dataset Overview

- **Initial Training Dataset**: Consists of 500 documents from PubMed and annotated with synthetic labels [Ref 10]. This dataset is used to train the base model (θ) before Active Learning begins.
- **Active Learning Data Pool**: Comprises 42 HDE labeled (Gold Standard data) documents, which are gradually introduced into the training process during Active Learning cycles. These documents are selected based on uncertainty measures and compared with the HDEs annotation results before being added to the updated training dataset.
- **Test Set**: Consists of 49 HDE labeled documents, which are never used in the training process and remain untouched throughout the Active Learning cycles. This dataset serves as the final benchmark for evaluating the model's performance, ensuring that improvements observed during Active Learning are not due to data leakage but reflect the model's genuine generalization capabilities.

  The HDE labeled documents were annotated by three medical domain experts and the Kappa rates were between 0.73 and 0.97. The details about the annotation have been published [Ref 10, 22, 23].

4.2 Metrics used in the experiments

In this study, we use three evaluation metrics to assess model performance: Recall and F1-score, and weighted F measure: fbeta_score [Ref 30]. These metrics are tracked across successive Active Learning cycles to measure how effectively the model improves in identifying key phrases as it incorporates human feedback.

Recall is selected as the primary metric of interest. This decision is motivated by the goal of minimizing the risk of missing any key phrases. A high recall indicates that the model successfully captures as many correct key phrases as possible, even if it occasionally includes false positives. F1-score provides a more holistic view of model performance and it balances both precision and recall. F1-score allows us to distinguish between strategies that achieve similar recall levels but differ in precision, offering a more nuanced comparison of Active Learning strategies. Therefore, we prioritize recall but also consider F1-score in this study. Weighted F measure is used to provide additional perspectives for readers. In this manuscript, we used beta equals 5 and 10, i.e., recall is 5 times or 10 times more important than precision [Ref 30].

4.3 Hyperparameters and Experiment Settings

To ensure fair and robust comparisons across all Active Learning strategies, we tuned the following **hyperparameters** systematically and compared the results. We systematically varied one hyperparameter at a time, while holding all others constant, to assess its impact on model performance. The best-performing values were then combined to form the final set of hyperparameters. Performance evaluation during this process was based on the test set.
The final experimental configurations include:

- **Number of Active Learning Cycles**: 4
- **Documents Added per Cycle**: 5
- **Number of Copies per Document per Cycle**: 1
- **BiLM Retrained**: Yes, the Language Model (BiLM) is retrained after each Active Learning cycle
- **Sequence Length (BiLM)**: 10
- **Learning Rate**: 0.001
- **Dropout Rate**: 0.3
- **Repetitions**: To ensure the stability and reliability of our results, we repeated each experiment 25 times and reported the average recall and F1 score along with their 95% confidence intervals (mean ± 1.96 × standard error). This approach mitigates variations across experiments and allows for a more meaningful comparison among different strategies.

The number of Active Learning (AL) cycles is a key hyperparameter that significantly influences model improvement over time. Typically, AL continues until a predefined stopping criterion is satisfied [Ref 24]. In our study, we initially experimented with eight cycles but found that most performance gains occurred during the early stages, with subsequent cycles yielding diminishing returns. Based on this observation, we chose to include four cycles of active learning and add five documents per cycle.

Additionally, informed by our previous experiments [Ref 10], we found that incorporating 1–2 HDE-annotated documents for every 100 synthetically labeled documents yields optimal results when HDE annotations are limited. Adding five HDE-annotated documents per cycle led to more stable and meaningful improvements, whereas adding only one often resulted in noisy or minimal gains. Given these considerations and a total budget of 20 HDE-annotated documents for AL, we naturally settled on four cycles.

Out of the 42 HDE-labeled documents available in our Active Learning pool, we allocated 20 for AL and reserved the remaining 22 to maintain variability in document selection across different strategies. This was a critical design choice—had all 42 documents been used for each strategy, the experiments would have become deterministic, limiting our ability to compare selection behavior. By limiting each strategy to 20 documents, we ensured diversity in selections, enabling fairer and more informative comparisons across Active Learning strategies.

4.4 Results

4.4.1 Evaluation results by different active learning strategies

The results presented here aim to evaluate how different strategies impact the rate and magnitude of these improvements, particularly in the context of recall. The average Recall and average F1-score metrics, along with their 95% confidence intervals, were reported for each Active Learning cycle and strategy in tables 3, 4, and 5. The strategy number corresponds to the combination lists in Table 2.

Table 3: Comparison of Token Entropy-based Active Learning strategies across four cycles (θ□–θ□, θ□ refers to the initial model trained on 500 synthetically labeled documents).

Evaluation results for token entropy (arithmetic average ± 1.96 standard error; n = 25)								
Active learning cycle	**DOCAVG REV ORDER**		**DOCAVG ACTUAL ORDER**		**KPSUM REV ORDER**		**KPSUM ACTUAL ORDER**	
	RECALL	**F1**	**RECALL**	**F1**	**RECALL**	**F1**	**RECALL**	**F1**
θ□	0.431 ± 0.0228	0.534 ± 0.0180	0.433 ± 0.0237	0.537 ± 0.0189	0.428 ± 0.0189	0.536 ± 0.0139	0.439 ± 0.0177	0.543 ± 0.0110
θ□	0.532 ± 0.0098	0.574 ± 0.0052	0.527 ± 0.0128	**0.576 ± 0.0053**	0.518 ± 0.0162	0.572 ± 0.0069	0.530 ± 0.0060	0.576 ± 0.0028
θ□	0.517 ± 0.0154	0.568 ± 0.0101	**0.538 ± 0.0152**	0.576 ± 0.0076	**0.539 ± 0.0138**	0.577 ± 0.0081	0.525 ± 0.0103	0.577 ± 0.0049
θ□	**0.536 ± 0.0164**	**0.578 ± 0.0109**	0.529 ± 0.0137	0.570 ± 0.0076	0.535 ± 0.0107	**0.581 ± 0.0077**	**0.544 ± 0.0091**	**0.586 ± 0.0056**
θ□	0.518 ± 0.0120	0.572 ± 0.0072	0.518 ± 0.0180	0.571 ± 0.0124	0.521 ± 0.0163	0.570 ± 0.0092	0.535 ± 0.0103	0.586 ± 0.0058

Across all four strategies—DOCAVG (actual and reverse), KPSUM (actual and reverse)—we observe consistent improvements in both Recall and F1 over the Active Learning cycles. The DOCAVG actual order and KPSUM actual order strategies show slightly higher final F1 scores (0.586) by θ_3, suggesting better overall generalization. KPSUM reverse order, although improving recall from θ□ to θ□, shows relatively lower F1 gains compared to the actual order strategies. Overall, the trends indicate that actual order sampling tends to be more effective than reverse order, and both DOCAVG and KPSUM aggregation methods yield comparable performance improvements over time.

All four Margin-based strategies show consistent improvement in recall and F1 scores from θ□ to θ□, with most gains occurring early in the Active Learning process. The DOCAVG (Actual Order) and KPSUM (Actual Order) strategies achieve the highest Recall scores by θ□, indicating that selecting documents in actual uncertainty order leads to more effective model updates. In contrast, reverse order strategies such as KPSUM (Reverse) and DOCAVG

(Reverse) show slower or plateaued improvement beyond θ□, suggesting that selecting less uncertain documents is less beneficial.

Table 4: Performance of Margin-based Active Learning strategies across four cycles (θ□–θ□, θ□ is the baseline model trained on 500 synthetically labeled documents).

Evaluation results for Margin (arithmetic average ± 1.96 standard error; n = 25)								
	DOCAVG REV ORDER		**DOCAVG ACTUAL ORDER**		**KPSUM REV ORDER**		**KPSUM ACTUAL ORDER**	
	RECALL	**F1**	**RECALL**	**F1**	**RECALL**	**F1**	**RECALL**	**F1**
θ□	0.43 ± 0.0159	0.539 ± 0.0110	0.446 ± 0.0138	0.548 ± 0.0086	0.439 ± 0.0115	0.542 ± 0.0088	0.440 ± 0.0137	0.546 ± 0.0084
θ□	0.522 ± 0.0123	**0.574 ± 0.0050**	0.520 ± 0.0146	0.574 ± 0.0089	0.502 ± 0.0153	0.555 ± 0.0110	0.522 ± 0.0110	**0.575 ± 0.0061**
θ□	**0.535 ± 0.0126**	0.573 ± 0.0085	**0.533 ± 0.0147**	0.578 ± 0.0059	0.518 ± 0.0155	0.557 ± 0.0097	**0.534 ± 0.0222**	0.573 ± 0.0137
θ□	0.525 ± 0.0148	0.573 ± 0.0076	0.530 ± 0.0092	**0.579 ± 0.0039**	**0.532 ± 0.0168**	**0.558 ± 0.0112**	0.525 ± 0.0144	0.572 ± 0.0094
θ□	0.522 ± 0.0156	0.573 ± 0.0098	0.525 ± 0.0125	0.577 ± 0.0058	0.515 ± 0.0141	0.557 ± 0.0103	0.528 ± 0.0154	0.575 ± 0.0098

Table 5: Performance of Minimum Token Probability (MTP)-based Active Learning strategies across five cycles (θ□–θ□).

Evaluation results for MTP (arithmetic average ± 1.96 standard error; n = 25)												
	DOCAVG REV ORDER		**DOCAVG ACTUAL ORDER**		**KPSUM REV ORDER**		**KPSUM ACTUAL ORDER**		**KPAVG REV ORDER**		**KPAVG ACTUAL ORDER**	
	RECALL	**F1**	**RECALL**	**F1**	**RECALL**	**F1**	**RECALL**	**F1**	**RECALL**	**F1**	**RECALL**	**F1**
θ□	0.45 ± 0.0166	0.546 ± 0.0099	0.435 ± 0.015	0.542 ± 0.0094	0.475 ± 0.0121	0.560 ± 0.0073	0.438 ± 0.0177	0.543 ± 0.0110	0.423 ± 0.0198	0.532 ± 0.0144	0.424 ± 0.0209	0.530 ± 0.0159
θ□	0.520 ± 0.0140	0.573 ± 0.0067	0.518 ± 0.0122	0.572 ± 0.0072	0.525 ± 0.0163	0.570 ± 0.0095	0.526 ± 0.0060	0.577 ± 0.0028	0.524 ± 0.0150	0.574 ± 0.0076	0.522 ± 0.0140	0.569 ± 0.0106
θ□	**0.543 ± 0.0203**	0.569 ± 0.0132	**0.546 ± 0.0145**	**0.583 ± 0.0066**	**0.540 ± 0.0185**	0.575 ± 0.0078	**0.543 ± 0.0103**	**0.586 ± 0.0049**	**0.538 ± 0.0132**	**0.578 ± 0.0056**	**0.532 ± 0.0152**	**0.574 ± 0.0085**
θ□	0.529	**0.579**	0.519	0.576	0.536	**0.576**	0.534	0.585	0.528	0.574	0.528	0.578

	± 0.0130	**± 0.0057**	± 0.0109	± 0.0066	± 0.0142	**± 0.0054**	± 0.0091	± 0.0056	± 0.0115	± 0.0058	± 0.0102	± 0.0064
θ□	0.533 ± 0.0117	0.576 ± 0.0051	0.525 ± 0.0116	0.569 ± 0.0059	0.534 ± 0.0130	0.574 ± 0.0069	0.532 ± 0.0103	0.576 ± 0.0058	0.529 ± 0.0148	0.573 ± 0.0077	0.501 ± 0.0174	0.565 ± 0.0103

Across all strategies using Minimum Token Probability (MTP), performance improves steadily through θ_4, with notable early gains between θ_0 and θ_1. KPAVG (Actual Order) and KPSUM (Actual Order) achieve the highest F1 scores by θ_2–θ_3 (≈0.574 and ≈0.586 respectively), reflecting the effectiveness of selecting documents with high token-level uncertainty. KPAVG (Reverse) also performs surprisingly well in early rounds, showing recall gains from 0.423 to 0.529, indicating that uncertainty in non-keyphrase tokens may still offer training value. DOCAVG strategies show more moderate improvements, suggesting that averaging token uncertainty across all tokens (rather than aggregating only over keyphrases) may dilute signal quality.

Random sampling shows an initial improvement in both recall and F1 from θ□ to θ□, with recall peaking at θ□ (0.587) before declining (Table 6). This fluctuation suggests inconsistency in sampling quality, and the lack of targeted selection likely limits model improvement beyond early rounds. F1 remains relatively stable after θ□, indicating general knowledge gain but lower efficiency compared to uncertainty-based strategies. The random sampling results serve as a baseline for evaluating the effectiveness of the active learning methods.

Table 6: Performance of the Random Sampling baseline across four active learning cycles (θ□–θ□).

RANDOM (arithmetic average ± 1.96 standard error; n = 25)		
	RECALL AVG	**F1 AVG**
θ□	0.44 ± 0.0135	0.543 ± 0.0098
θ□	0.523 ± 0.0135	**0.579 ± 0.0057**
θ□	**0.587 ± 0.0108**	0.577 ± 0.0047
θ□	0.519 ± 0.0135	0.571 ± 0.0088

θ□	0.509 ± 0.0172	0.574 ± 0.0112

4.4.2 Evaluation results comparison across different active learning strategies

Figure 3a, 3b, 3c presents the Recall improvement across Active Learning cycles for all strategies, including confidence intervals. A notable trend across nearly all strategies is that the majority of performance improvement occurs between the initial model (θ□) and the first cycle (θ□). Towards the later cycles, most strategies exhibit fluctuations in recall, with little to no consistent upward trend, suggesting that performance gains begin to plateau.

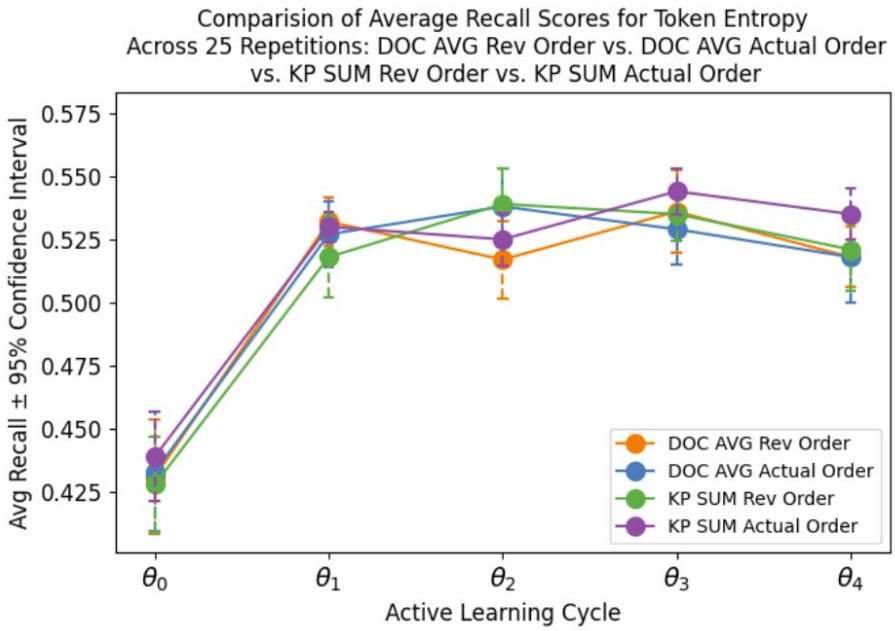

Figure 3(a): Recall improvement across Active Learning cycle for Token Entropy

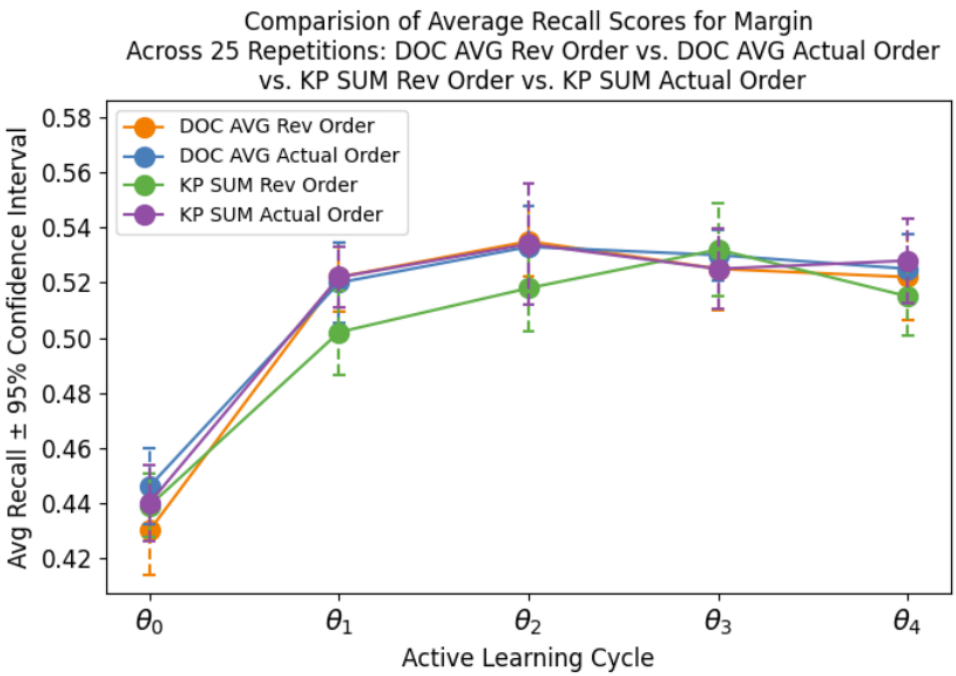

Figure 3(b): Recall improvement across Active Learning cycle for Margin

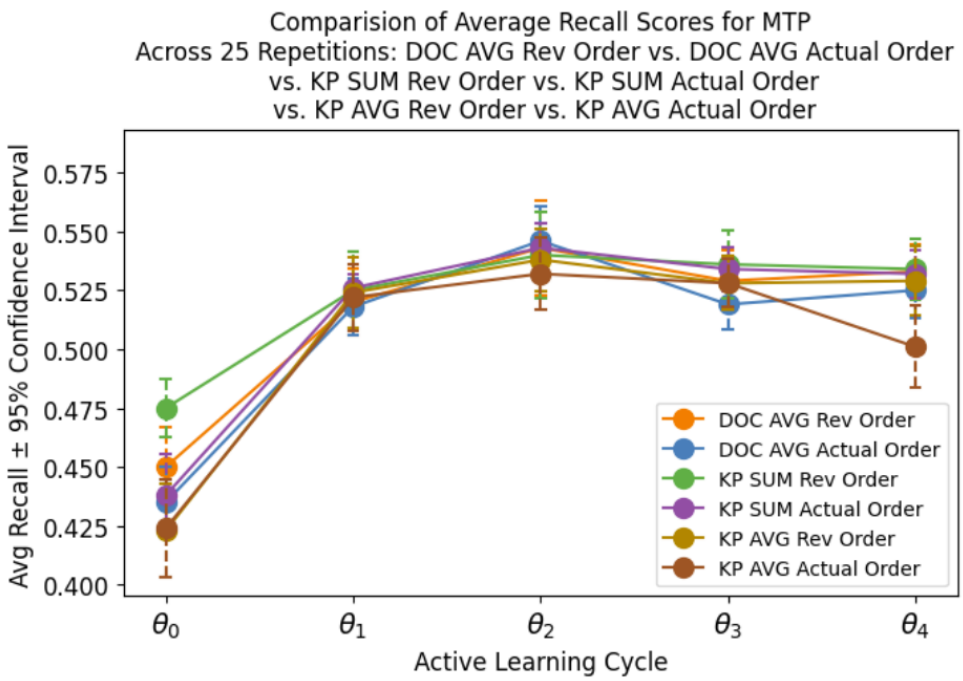

Figure 3(c): Recall improvement across Active Learning cycle for MTP

Figure 3: Recall improvement across Active Learning cycles for all strategies

Figure 4 shows the F1 score progression across Active Learning cycles for all strategies, along with confidence intervals. Similar to the recall trends in Figure 3, most of the improvement in F1 score is observed between the initial model (θ□) and the first cycle (θ□). In the subsequent cycles, the F1 scores for most strategies fluctuate, indicating that performance stabilizes and significant improvements become less frequent over time. Noticeably, our results demonstrate a slightly larger improvement for recall than for F1 scores between θ□ and θ□.

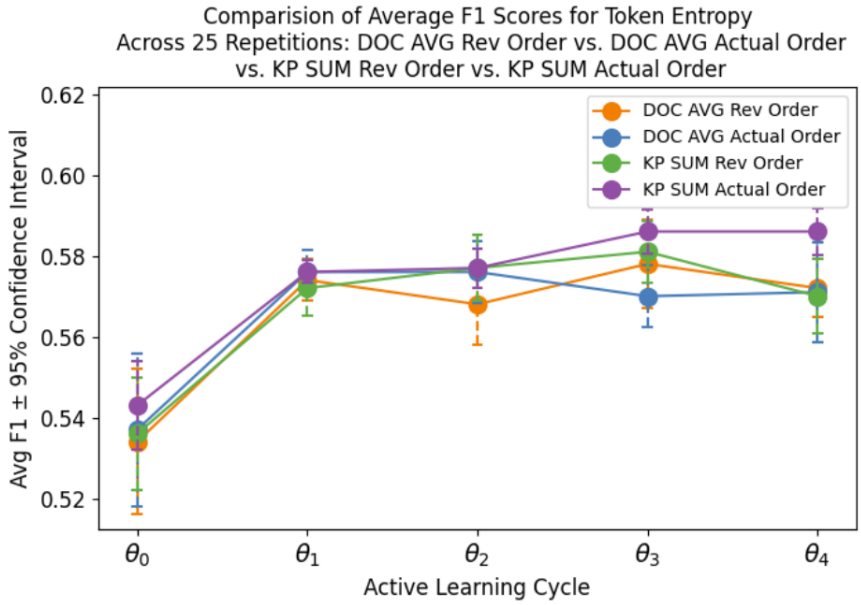

Figure 4(a): F1 score progression across Active Learning cycles for Token Entropy

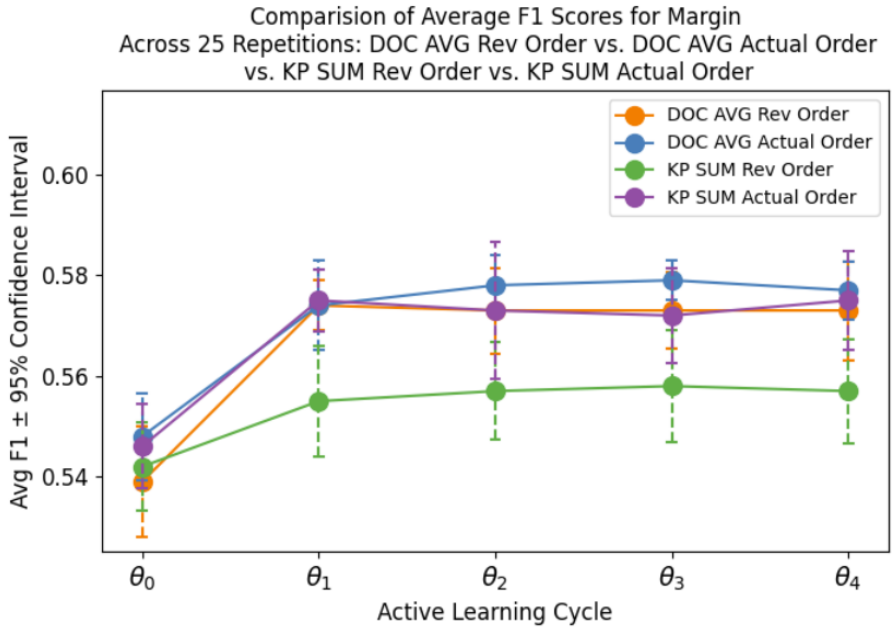

Figure 4(b): F1 score progression across Active Learning cycles for Margin

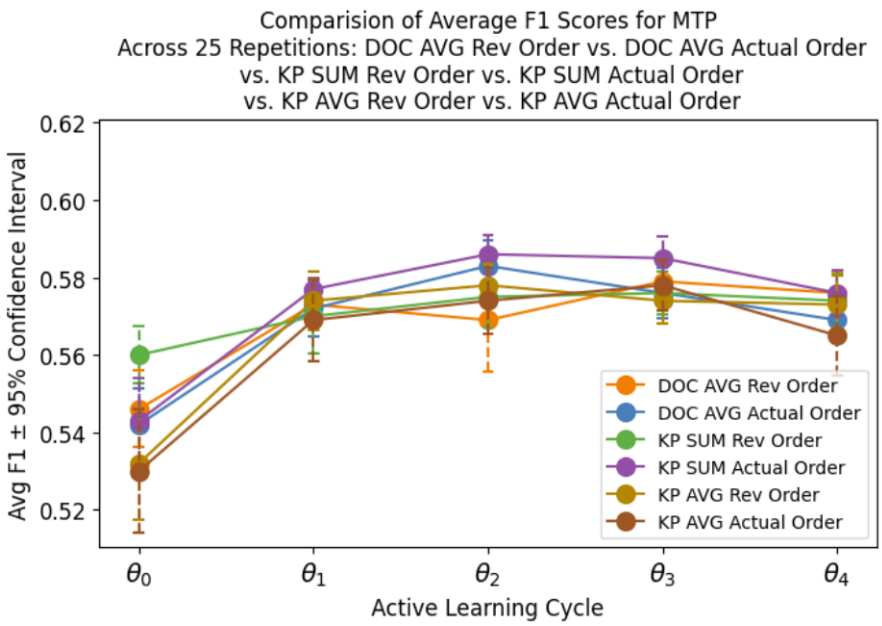

Figure 4(c): F1 score progression across Active Learning cycles for MTP

Figure 5 compares the average Recall scores for the MTP KPSum Reverse and MTP KPSum Actual strategies across Active Learning cycles, including their confidence intervals. Both strategies demonstrate notable improvements in recall, with KPSum Reverse showing consistent performance and steady improvements throughout the cycles. The recall starts at 0.475 ± 0.0121 in θ□ and peaks at 0.540 ± 0.0185 in θ□. On the other hand, KPSum Actual starts lower at 0.438 ± 0.0177 but surpasses KPSum Reverse after θ□, eventually reaching 0.532 ± 0.0103 in θ□. It reaches a peak of 0.543 ± 0.0103 in θ□. Although both strategies improve in recall over time, KPSum Actual exhibits a more pronounced performance improvement, particularly in the later cycles.

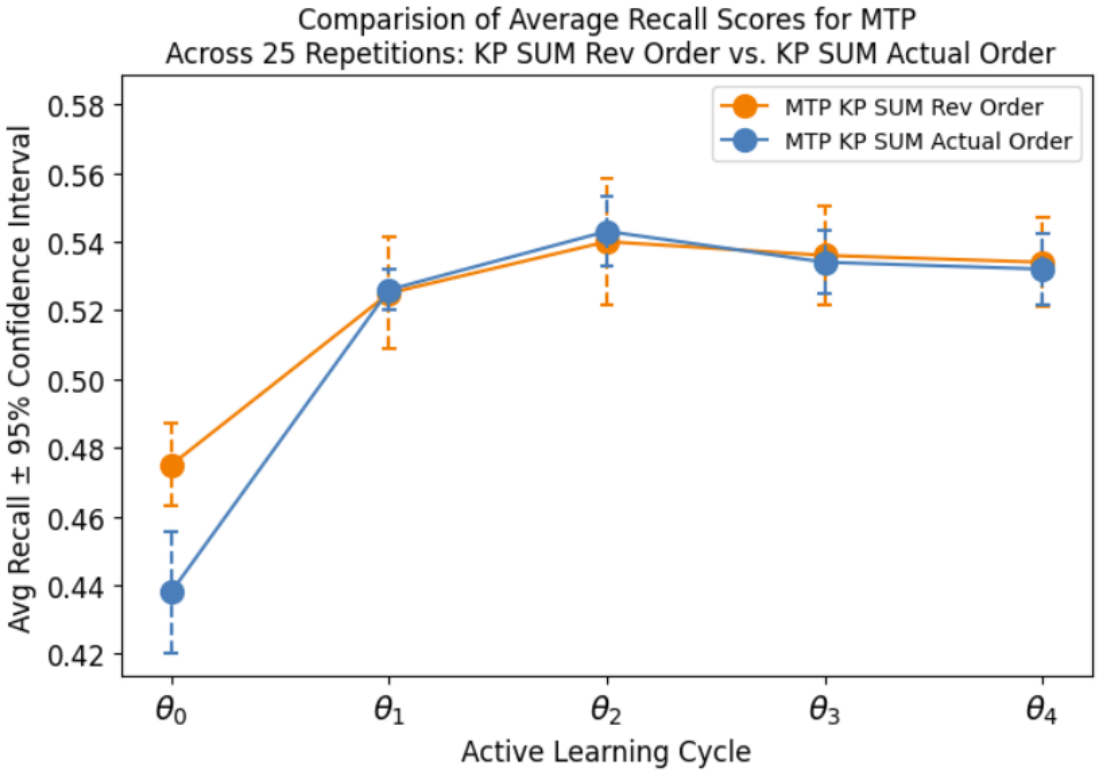

Figure 5: Recall score comparison between MTP KPSum reverse and MTP KPSum Actual

Figure 6 illustrates the comparison of F1 scores between MTP KPSum Reverse and MTP KPSum Actual strategies over the course of Active Learning cycles. Both strategies show similar performance trends in terms of F1 scores, with KPSum Reverse starting with 0.560 ± 0.0073 in θ□ and reaching a peak of 0.576 ± 0.0054 in θ□. KPSum Actual starts slightly lower at 0.543 ± 0.0110 but improves steadily, surpassing KPSum Reverse in later cycles to reach 0.576 ± 0.0058 in θ□. It reaches a peak of 0.586 ± 0.0049 in θ□. The performance of both strategies stabilizes as they approach the saturation point, with KPSum Actual maintaining a consistent and slightly higher F1 score towards the end, indicating the effectiveness of the actual order strategy in capturing keyphrases.

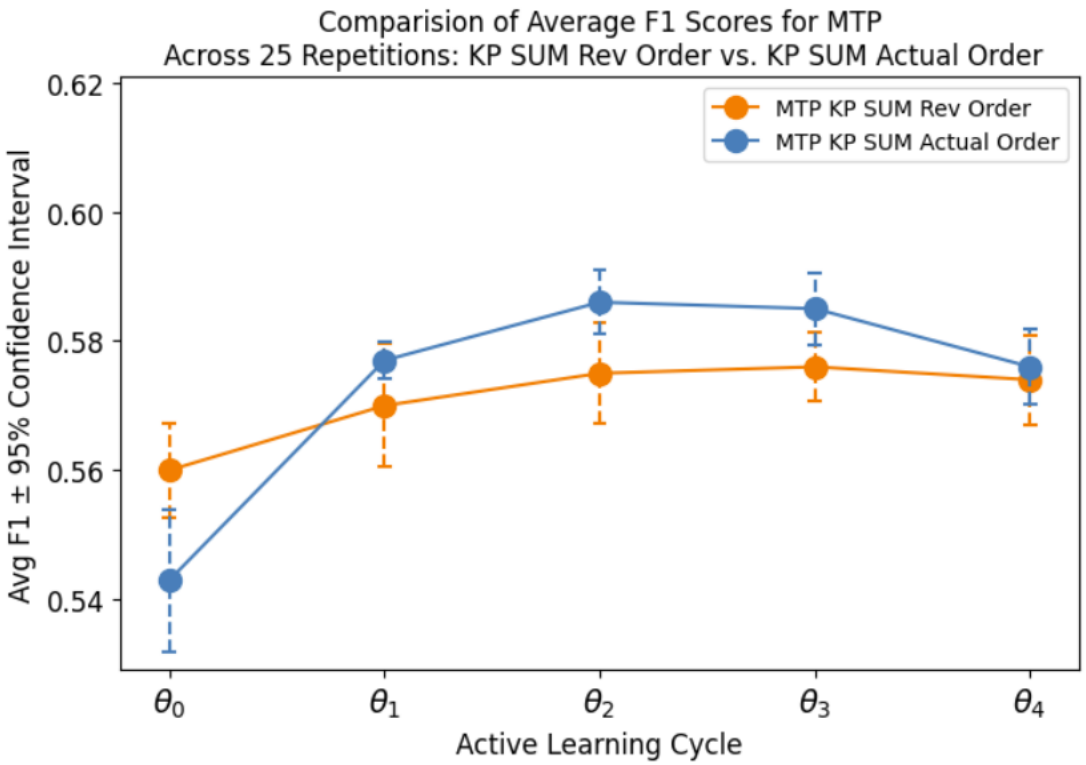

Figure 6: F1 score comparison between MTP KPSum reverse and MTP KPSum Actual

Figure 7 compares the average F1 scores between the MTP KpAvg Actual and MTP KpAvg Reverse strategies across Active Learning cycles, including their respective confidence intervals. Both strategies show similar trends in performance improvement across cycles. In the initial cycles, both strategies exhibit close F1 values, with KpAvg Actual showing a slight edge in later cycles, reaching 0.578 ± 0.0064 in θ□. On the other hand, KpAvg Reverse reaches a peak at 0.578 ± 0.0056 in θ□, demonstrating consistent performance. This indicates that while both strategies exhibit steady performance, the reverse strategy yields slightly better results in the later cycles.

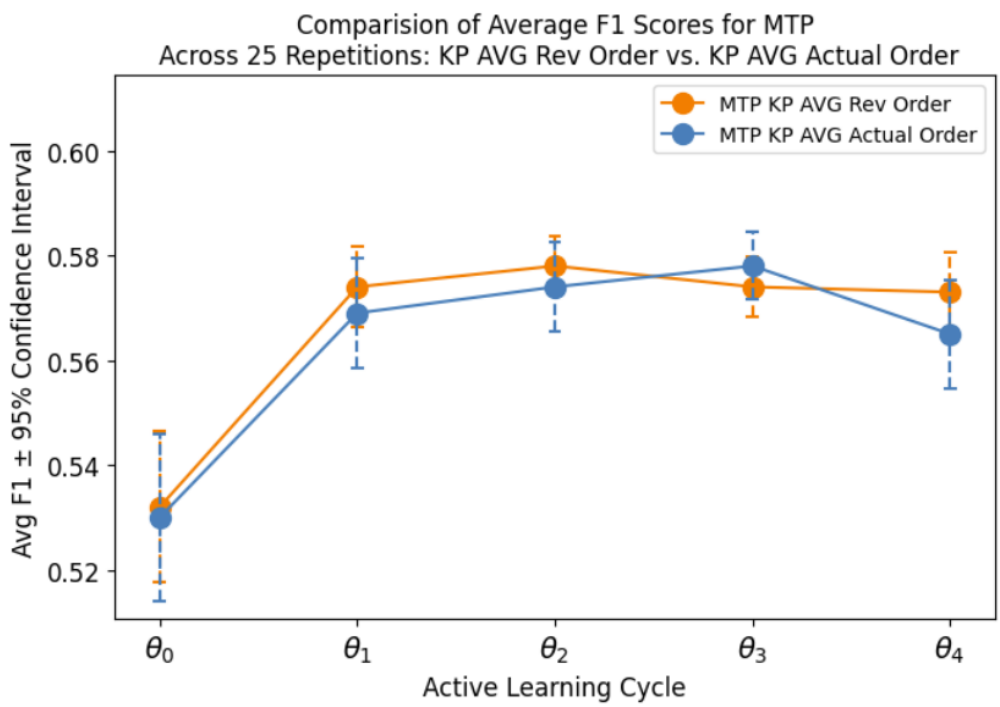

Figure 7: Average F1 comparison of MTP KpAvg Reverse and MTP KpAvg Actual Strategies.

Figure 8 shows the average Recall scores for MTP KpAvg Actual and MTP KpAvg Reverse strategies over Active Learning cycles, along with their confidence intervals. Similar to the F1 comparison, MTP KpAvg Actual and MTP KpAvg Reverse show comparable Recall values in the earlier cycles. However, KpAvg Reverse demonstrates slightly better recall, peaking at 0.538 ± 0.0132 in θ_2, while KpAvg Actual reaches 0.532 ± 0.0152 in the same cycle. Both strategies exhibit a slight decline in recall towards the final cycle, with KpAvg Reverse maintaining a small advantage over KpAvg Actual. This observation reinforces the idea that the reverse strategy, while showing some fluctuations, yields more stable and better Recall in the later cycles.

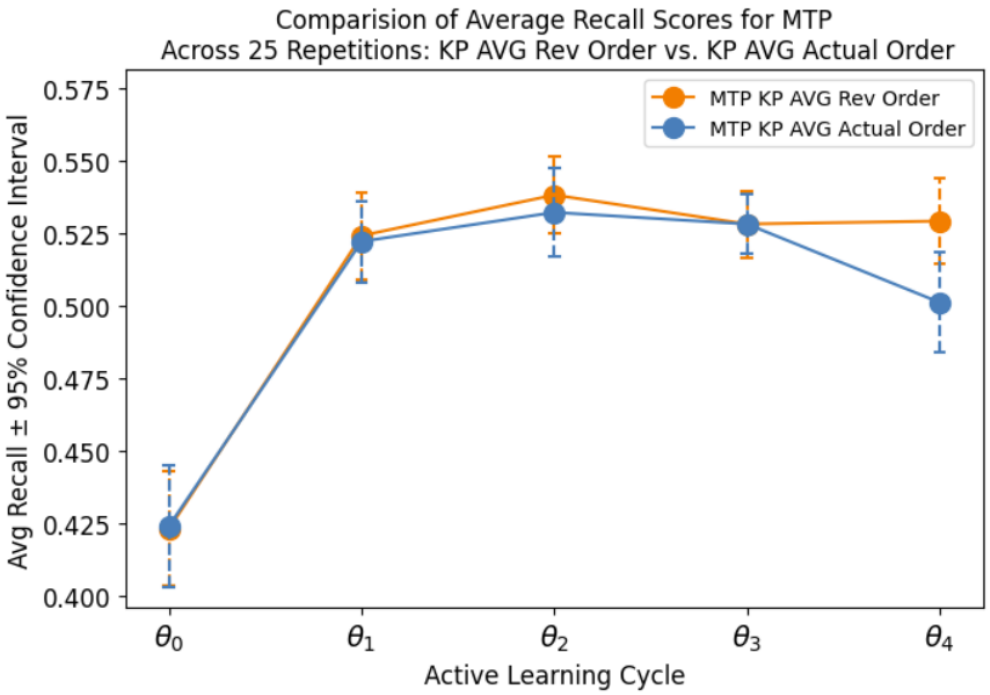

Figure 8: Average Recall comparison of MTP KpAvg Reverse and MTP KpAvg Actual Strategy.

Figure 9 illustrates the average Recall performance across Active Learning cycles for the Token Entropy KPSum Actual strategy compared to Random sampling, along with their respective confidence intervals. Both strategies start with nearly identical performance at θ_0 (≈ 0.44). In the early cycles, Random shows a sharper increase in recall, peaking at θ_2 with 0.587 ± 0.0108, while KPSum Actual steadily improves and reaches 0.544 ± 0.0091 at its peak in θ_3. However, in later cycles, Random sampling exhibits a noticeable drop in performance, declining to 0.509 ± 0.0172 at θ_4. In contrast, KPSum Actual maintains a more stable trend, with only a slight dip in the final cycle (0.535 ± 0.0103), suggesting more consistent and reliable performance over time. This pattern highlights the tendency of Random selection to produce early but less stable gains, while Active Learning via KPSum Actual leads to more gradual but sustained improvements.

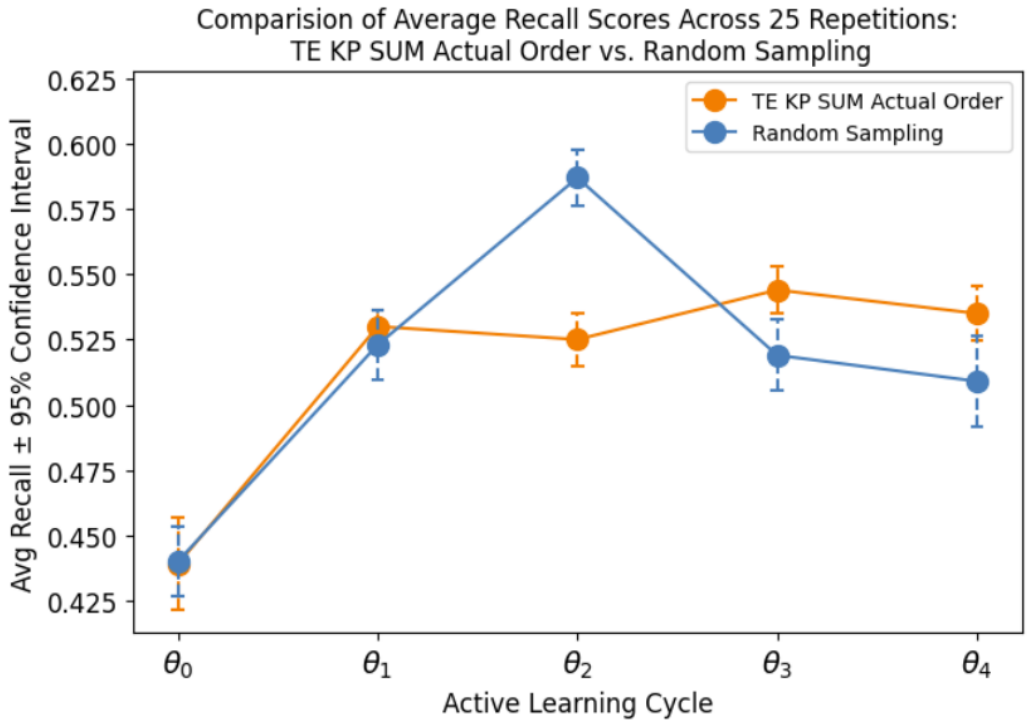

Figure 9: Average Recall comparison of TE KpSum Actual and Random Sampling

Figure 10 compares the average F1 scores across Active Learning cycles between the Token Entropy KPSum Actual strategy and Random sampling, with 95% confidence intervals included. At the starting point (θ□), both strategies show nearly identical F1 performance (~0.543). Random sampling demonstrates a noticeable increase in the early cycles, peaking at 0.579 ± 0.0057 by θ□, followed by a gradual decline through later cycles. In contrast, KPSum Actual shows a smaller but steadier improvement, reaching 0.586 ± 0.0056 by θ□ and maintaining that level in θ□ (0.586 ± 0.0058). This figure highlights a key trend: while Random sampling can initially outperform in terms of raw F1 gains, its performance tends to fluctuate more in later cycles. On the other hand, the KPSum Actual strategy offers more consistent and stable improvements over time.

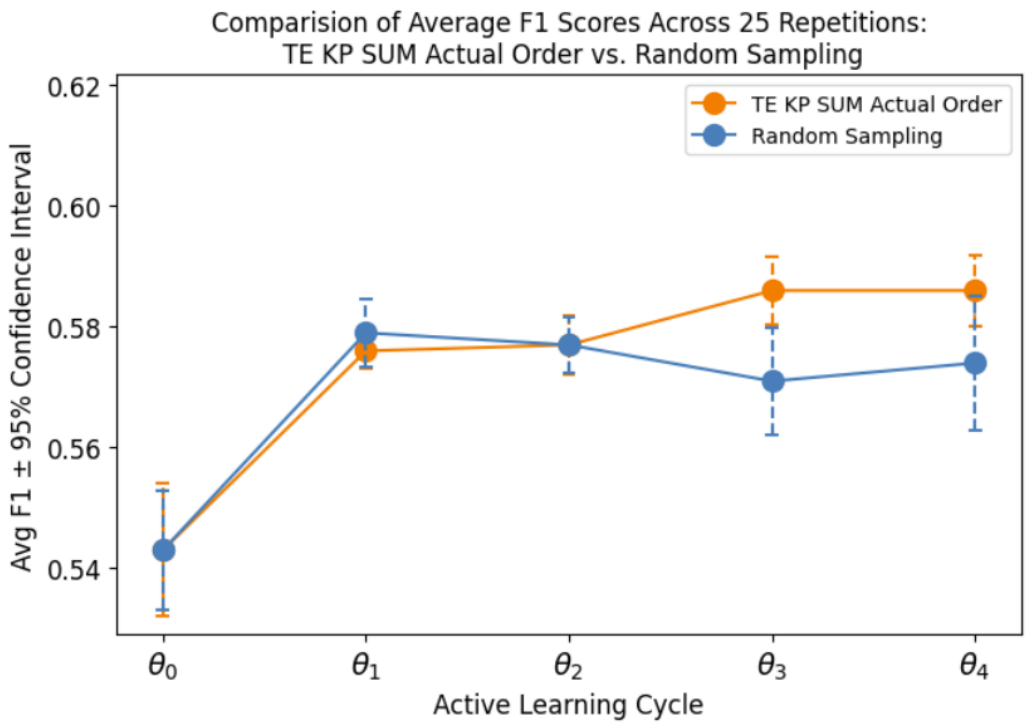

Figure 10: Average F1 comparison of TE Kpsum Actual and Random Sampling

4.4.3 Original and weighted F1 values by different active learning strategies

Table 7 reports the weighted F1 scores (± 1.96 standard error) for Token Entropy–based Active Learning strategies evaluated at β = 1, β = 5, and β = 10 across five AL cycles (θ□ to θ□). At θ□, all strategies start with comparable scores, ranging from 0.430 to 0.445. DOCAVG Actual Order shows consistent performance, reaching 0.542 (β = 5) at θ□. KPSUM Actual Order attains the highest score of 0.546 (β = 5) at θ□, while also maintaining stable results across other cycles. KPSUM Reverse Order peaks at θ□ with 0.542 (β = 5) and slightly declines in later cycles. DOCAVG Reverse Order sees a rise from 0.439 (θ□, β = 5) to 0.541 (θ□, β = 5), then slightly fluctuates. Across all strategies and cycles, the β = 5 and β = 10 values remain closely aligned, differing only marginally within each configuration.

Table 7: Weighted F1 scores for Token Entropy–based Active Learning strategies, evaluated at β = 1, β = 5, and β = 10.

COMBINED WEIGHTED F1 FOR TOKEN ENTROPY (arithmetic average ± 1.96 standard error; n = 25)												
	DOCAVG REV ORDER			**DOCAVG ACTUAL ORDER**			**KPSUM REVERSE ORDER**			**KPSUM ACTUAL ORDER**		
	Beta = 1	**Beta = 5**	**Beta = 10**	**Beta = 1**	**Beta = 5**	**Beta = 10**	**Beta = 1**	**Beta = 5**	**Beta = 10**	**Beta = 1**	**Beta = 5**	**Beta = 10**
θ□	0.534 ± 0.0180	0.439 ± 0.0227	0.434 ± 0.0228	0.537 ± 0.0189	0.440 ± 0.0236	0.435 ± 0.0237	0.536 ± 0.0139	0.434 ± 0.0186	0.430 ± 0.0188	0.543 ± 0.0110	0.445 ± 0.0174	0.440 ± 0.0177
θ□	0.574 ± 0.0052	0.523 ± 0.0094	0.52 ± 0.0097	**0.576 ± 0.0053**	0.529 ± 0.0121	0.526 ± 0.0126	0.572 ± 0.0069	0.522 ± 0.0156	0.519 ± 0.0160	0.576 ± 0.0028	0.523 ± 0.0058	0.527 ± 0.0060
θ□	0.568 ± 0.0101	**0.541 ± 0.0150**	**0.539 ± 0.0153**	0.576 ± 0.0076	**0.542 ± 0.0146**	0.526 ± 0.0150	0.577 ± 0.0081	**0.542 ± 0.0134**	**0.540 ± 0.0137**	0.577 ± 0.0049	**0.546 ± 0.0099**	**0.544 ± 0.0102**
θ□	**0.578 ± 0.0109**	0.523 ± 0.0160	0.521 ± 0.0163	0.570 ± 0.0076	0.531 ± 0.0132	**0.529 ± 0.0136**	**0.581 ± 0.0077**	0.539 ± 0.0104	0.536 ± 0.0106	**0.586 ± 0.0056**	0.538 ± 0.0088	0.535 ± 0.0090
θ□	0.572 ± 0.0072	0.535 ± 0.0116	0.533 ± 0.0119	0.571 ± 0.0124	0.520 ± 0.0177	0.517 ± 0.0179	0.570 ± 0.0092	0.525 ± 0.0157	0.522 ± 0.0162	0.586 ± 0.0058	0.535 ± 0.0099	0.533 ± 0.0102

Table 8 presents the weighted F1 scores (± 1.96 standard error) for Margin-based Active Learning strategies across five AL cycles (θ□ to θ□), evaluated at β = 1, β = 5, and β = 10. At θ□. All strategies show a marked improvement by θ□, with scores reaching above 0.52 across the board. DOCAVG Actual Order and KPSUM Actual Order both reach a peak F1 score of 0.536 at θ□ (β = 5), while other strategies maintain competitive values in the 0.519–0.537 range. Standard errors remain consistent across cycles and strategies, suggesting stable performance. The values for β = 5 and β = 10 remain closely aligned throughout, with minimal variance between them in any given configuration.

Table 8: Weighted F1 scores for Margin–based Active Learning strategies, evaluated at β = 1, β = 5, and β = 10.

COMBINED WEIGHTED F1 FOR MARGIN (arithmetic average ± 1.96 standard error; n = 25)												
	DOCAVG REV ORDER			**DOCAVG ACTUAL ORDER**			**KPSUM REVERSE ORDER**			**KPSUM ACTUAL ORDER**		
	Beta = 1	**Beta = 5**	**Beta = 10**	**Beta = 1**	**Beta = 5**	**Beta = 10**	**Beta = 1**	**Beta = 5**	**Beta = 10**	**Beta = 1**	**Beta = 5**	**Beta = 10**
θ□	0.539 ± 0.0110	0.436 ± 0.0157	0.431 ± 0.0159	0.548 ± 0.0086	0.452 ± 0.0139	0.447 ± 0.0141	0.542 ± 0.0088	0.445 ± 0.0112	0.440 ± 0.0113	0.546 ± 0.0084	0.447 ± 0.0135	0.442 ± 0.0136
θ□	**0.574 ± 0.0050**	0.526 ± 0.0118	0.523 ± 0.0122	0.574 ± 0.0089	0.524± 0.0145	0.521 ± 0.0148	0.555 ± 0.0110	0.506 ± 0.0147	0.503 ± 0.0149	**0.575 ± 0.0061**	0.526 ± 0.0106	0.523 ± 0.0109
θ□	0.573 ± 0.0085	**0.537 ± 0.0122**	**0.535± 0.0125**	0.578 ± 0.0059	**0.536 ± 0.0144**	**0.534 ± 0.0149**	0.557 ± 0.0097	**0.521 ± 0.0153**	**0.519 ± 0.0157**	0.573 ± 0.0137	**0.536 ± 0.0216**	**0.534 ± 0.0221**
θ□	0.573 ± 0.0076	0.528 ± 0.0143	0.526 ± 0.0147	**0.579 ± 0.0039**	0.534 ± 0.0090	0.531 ± 0.0093	**0.558 ± 0.0112**	0.509 ± 0.0162	0.506 ± 0.0165	0.572 ± 0.0094	0.528 ± 0.0141	0.526 ± 0.0143
θ□	0.573 ± 0.0098	0.526 ± 0.0152	0.523 ± 0.0155	0.577 ± 0.0058	0.528 ± 0.0122	0.526 ± 0.0126	0.557 ± 0.0103	0.518 ± 0.0135	0.516 ± 0.0137	0.575 ± 0.0098	0.531 ± 0.0150	0.529 ± 0.0153

Table 9 displays weighted F1 scores (± 1.96 standard error) for various MTP-based Active Learning strategies over five AL cycles (θ□ to θ□), evaluated at β = 1, β = 5, and β = 10. All configurations show clear performance improvements by θ□, with F1 scores converging around 0.521–0.529 for β = 5, and β = 10. Peak values are generally observed at θ□. From θ□ to θ□, scores mostly stabilize, with only minor variations across β values. KPAVG Actual sees a slight drop at θ□ (0.502 at β = 10), while other strategies maintain consistency. Overall, standard errors remain low, indicating stable and repeatable performance across AL cycles and β settings.

Table 9: Weighted F1 scores for MTP–based Active Learning strategies, evaluated at β = 1,5 and 10.

COMBINED WEIGHTED F1FOR MTP (arithmetic average ± 1.96 standard error; n = 25)												
	DOCAVG REV ORDER			**DOCAVG ACTUAL ORDER**			**KPSUM REVERSE ORDER**			**KPSUM ACTUAL ORDER**		
	Beta = 1	**Beta = 5**	**Beta = 10**	**Beta = 1**	**Beta = 5**	**Beta = 10**	**Beta = 1**	**Beta = 5**	**Beta = 10**	**Beta = 1**	**Beta = 5**	**Beta = 10**
θ□	0.546 ± 0.0099	0.456 ± 0.0166	0.451 ± 0.0169	0.542 ± 0.0094	0.442 ± 0.0148	0.437 ± 0.0150	0.560 ± 0.0073	0.480 ± 0.0118	0.476 ± 0.0121	0.543 ± 0.0110	0.445 ± 0.0174	0.440 ± 0.0177
θ□	0.573 ± 0.0067	0.524 ± 0.0134	0.521 ± 0.0137	0.572 ± 0.0072	0.521± 0.0118	0.518 ± 0.0121	0.570 ± 0.0095	0.528 ± 0.0157	0.526 ± 0.0161	0.577 ± 0.0028	0.529 ± 0.0058	0.527 ± 0.0060
θ□	0.569 ± 0.0132	**0.545 ± 0.0204**	**0.543 ± 0.0209**	**0.583 ± 0.0066**	**0.548 ± 0.0139**	**0.547± 0.0143**	0.575 ± 0.0078	**0.542 ± 0.0176**	**0.541 ± 0.0182**	**0.586 ± 0.0049**	**0.546 ± 0.0099**	**0.544 ± 0.0102**
θ□	**0.579 ± 0.0057**	0.533 ± 0.0123	0.530 ± 0.0127	0.576 ± 0.0066	0.523 ± 0.0106	0.520 ± 0.0108	**0.576 ± 0.0054**	0.539 ± 0.0132	0.537 ± 0.0139	0.585 ± 0.0056	0.538 ± 0.0088	0.535 ± 0.0090
θ□	0.576 ± 0.0051	0.535 ± 0.0113	0.533 ± 0.0119	0.569 ± 0.0059	0.528 ± 0.0112	0.525 ± 0.0115	0.574 ± 0.0069	0.536 ± 0.0124	0.534 ± 0.0129	0.576 ± 0.0058	0.535 ± 0.0099	0.533 ± 0.0102

Table 10 reports the weighted F1 scores (± 1.96 standard error) for the Random Active Learning strategy evaluated at β = 1, β = 5, and β = 10. The model begins at θ□ with scores of 0.446 (β = 5) and 0.441 (β = 10). There is a steady increase through θ□ and θ□, peaking at θ□ with the highest scores: 0.586 (β = 5) and 0.587 (β = 10). This indicates that Random selection can still significantly enhance performance in early AL cycles. However, a decline follows, with scores dropping to 0.523 (β = 5) and 0.520 (β = 10) at θ□, and further to 0.514 and 0.510 at θ□. These results suggest that while Random sampling provides early performance gains, it lacks consistency in maintaining or improving performance in later stages of the Active Learning process.

Table 10: Weighted F1 scores for Random Strategy, evaluated at β = 1, β = 5, and β = 10.

RANDOM (arithmetic average ± 1.96 standard error; n = 25)			
	Beta = 1	**Beta = 5**	**Beta = 10**
θ□	0.543 ± 0.0098	0.446 ± 0.0133	0.441± 0.0134
θ□	**0.579 ± 0.0057**	0.527 ± 0.0129	0.524 ± 0.0134
θ□	0.577 ± 0.0047	**0.586 ± 0.0099**	**0.587 ± 0.0106**
θ□	0.571 ± 0.0088	0.523 ± 0.0127	0.520± 0.0130
θ□	0.574 ± 0.0112	0.514 ± 0.0168	0.510 ± 0.0171

5 Discussion

This section presents the observations and interpretation of the Active Learning experiment results, highlighting how different uncertainty measures, aggregation methods, and document selection orders affect model performance. Across all strategies, we observe that the majority of performance improvement occurs between θ□ and θ□, particularly in recall, which is the primary metric of this study. We also note that Token Entropy and MTP produce similar document rankings, leading to comparable outcomes across cycles.

Among all evaluated strategies, KP SUM Actual consistently leads to higher recall improvements in the early cycles, attributed to its preference toward selecting documents rich in key phrases. These documents not only help the model capture more key phrase patterns but also improve its general learning, reflected in gains in both recall and F1 scores. In contrast, KP SUM Reverse, while more stable, shows relatively smaller recall gains due to its tendency to select documents with fewer key phrases.

We also analyze F1 and recall dynamics in KP AVG strategies, noting that KP AVG Actual tends to prioritize uncertain key phrases and boosts recall early on, while KP AVG Reverse focuses on relatively more confident key phrases, helping reduce false positives and slightly improving F1. Finally, we identify saturation points in later cycles where gains taper off, and indicate stopping criteria to ensure efficient use of HDE labeling efforts. Based on these findings, we select KP SUM Actual as the most effective strategy for our use case, balancing both performance improvement and annotation efficiency.

5.1 Majority of Performance Gains Occur from θ_0 to θ_1.

A notable observation in our experiments is that the majority of performance improvement occurs during the transition from θ_0 (the initial model without active learning) to θ_1 (the first cycle of active learning) regardless of the strategies. This pattern aligns with findings from previous studies [Ref 25], which show that significant gains are often achieved in the early stages of active learning.

This behavior highlights the importance of the initial rounds of feedback, as they address the most critical gaps in the model's learning, leading to rapid improvements. It emphasizes the need to prioritize high-quality labels and diverse data during the initial cycles to maximize the impact of the active learning process.

Our use case differs from standard Active Learning scenarios in two crucial ways: the training data for θ_0 is purely synthetically labelled. As a result, when data with HDE labels are introduced in θ_1, the model gains exposure to the ground truth data patterns. This suggests that the performance improvement from θ_0 to θ_1 is not solely due to Active Learning's uncertainty-based document selection but also because the model is learning more representative key phrase structures from the HDE labelled data. Another unique characteristic of our use case is the complexity of the task, which is significantly different from a typical binary predictive task.

As supporting evidence, in figures 3, and 4, we observe that even when documents are added randomly, there is still a significant improvement from θ_0 to θ_1 in both F1 and recall metrics. This indicates that while Active Learning strategies optimize sample selection, the introduction of HDE labeled data itself is a major factor in the observed performance boost. Therefore, a substantial portion of the improvement can be attributed to the transition from synthetic labels to human-labeled data, reinforcing the importance of integrating HDE annotated documents early in the learning process to improve generalization.

5.2 Identifying Saturation in Active Learning Cycles:

Determining an effective stopping criterion is crucial for ensuring that Active Learning remains efficient while avoiding unnecessary annotations. A well-defined stopping condition prevents excessive labeling costs and mitigates the risk of overfitting the active learning model.

One of the most straightforward approaches is to monitor the convergence of learning by evaluating the generalization error on a test dataset. This approach, as discussed in H Ishibashi, et al *[Ref 24]*, provides a way to assess when learning has stabilized. However, obtaining a sufficiently large and representative test set may not always be feasible, making alternative stopping strategies necessary.

A practical alternative is to define saturation-based stopping criteria, where the Active Learning process halts once performance improvements fall below a predefined threshold (e.g., less than 1% improvement in recall or F1-score) or when key evaluation metrics begin to fluctuate around a stable value. Additionally, tracking model uncertainty over multiple cycles can help identify

when the most informative data points have already been selected, signaling a natural stopping point. By establishing clear stopping criteria, Active Learning can be conducted efficiently, ensuring that the process runs until meaningful gains are no longer observed.

As shown in most of the strategies presented in Tables 3, 4, and 5, and visualized in Figures 3 and 4, the recall generally increases from θ□ to θ□, after which it either plateaus or begins to fluctuate slightly without consistent gains. This trend suggests that the model makes its most substantial improvements during the early cycles of active learning. Specifically, by the end of θ□, the model appears to have incorporated the most informative and uncertain samples from the pool, allowing it to generalize better and detect a larger number of keyphrases. After this point, additional labeled data contribute only marginal improvements or noise, as the remaining samples are less helpful in distinguishing keyphrases from non-keyphrases. A similar trend is observed in F1 scores, which rise along with recall up to θ□ but tend to stabilize thereafter. This indicates that the balance between precision and recall no longer shifts significantly, reinforcing the idea that the model has reached a saturation point in learning from the remaining data. The convergence of both recall and F1 around θ□ supports the use of θ□ as a potential stopping point for the active learning process. Continuing beyond θ□ may lead to inefficiencies, both in terms of annotation effort and computational resources, without offering proportionate improvements in model performance.

However, it is important to note that we may be observing saturation earlier in our experiments due to the limited size of our Active Learning data pool. With only 42 documents available for Active Learning, the selection choices become increasingly restricted in later cycles, which may contribute to the observed plateau in recall and other performance metrics. In a larger Active Learning dataset, the expected saturation point may occur at a later stage, allowing for more extended model improvement.

By establishing clear stopping criteria, we ensure that Active Learning resources are utilized efficiently, stopping the experiment once meaningful gains are no longer observed while recognizing the limitations imposed by our restricted Active Learning data pool.

5.3 Impact of KP SUM Strategies on Recall and F1 Improvement in Active Learning

In Tables 3, 4, and 5, the most consistent and stable improvements in recall are observed for KPSUM Actual. Not only does this strategy show the highest recall gains across most uncertainty measures, but it also demonstrates lower variation in the confidence intervals, indicating more stable learning. While certain other strategies—especially those based on different uncertainty sampling methods—may outperform KPSUM Actual in isolated instances, none exhibit the same level of consistency across all metrics and sampling techniques.

The effectiveness of KPSUM lies in its construction: it calculates document uncertainty solely based on key phrases, assigning higher scores to documents with a greater density of such phrases. By prioritizing these keyphrase-rich documents during selection, the model is repeatedly exposed to examples that reinforce the structure, context, and patterns associated with key phrases. This focused exposure enables the model to more efficiently internalize what

constitutes a key phrase, leading to more robust generalization and improved recall performance.

Although the per-cycle recall gain of KPSUM Actual may not always be dramatically higher than other strategies, its sustained improvement and model stability suggest a long-term advantage. We believe that with a larger Active Learning pool, KPSUM Actual would likely continue to outperform other methods with greater distinction.

Figures 3 and 4 further illustrate this trend. Specifically, when comparing MTP KPSUM Actual with MTP DOCAVG Actual, MTP KPAVG Actual, and Random sampling, Figure 4 shows that F1 scores for KPSUM Actual improve gradually and consistently across cycles. This highlights its strength in maintaining steady performance gains. Likewise, in Figure 3, the recall trajectory for KPSUM Actual is not only higher but also smoother, with minimal fluctuations. This stability underscores the ability of KPSUM Actual to guide the model toward meaningful, reliable improvements over time—particularly in capturing key phrase patterns.

The consistent improvement in F1 scores for KP SUM Actual, as shown in Figure 6, may also be attributed to its tendency to select longer documents. These documents naturally contain more tokens and offer richer context, enabling the model to learn more effectively to distinguish between key phrases and non-key phrases. In contrast, KP SUM Reverse often selects shorter documents with fewer tokens and less contextual variety, leading to reduced generalization and minimal F1 improvements.

Another key observation in our experiment results is that the recall improvement from θ□ to θ□ in the KPSUM Actual strategy is consistently greater than in KPSUM Reverse in most cases, particularly in Tables 3 and 5, which correspond to the Token Entropy and MTP uncertainty measures. In Figure 5, where the MTP uncertainty measure is applied, this pattern is clearly visible. This suggests that selecting documents in their actual order tends to result in more informative updates during the early stages of learning compared to reverse-order selection. One possible explanation is that KPSUM Actual prioritizes keyphrase-rich documents in a manner that aligns more naturally with the dataset's distribution, enabling the model to encounter and learn from more representative and informative examples early on.

However, an exception is observed in Table 3, which uses the Margin uncertainty measure. In this case, KPSUM Reverse shows a slightly greater improvement from θ□ to θ□ than KPSUM Actual, though the difference is small and KPSUM Actual still remains close behind. This suggests that while actual-order selection is generally advantageous, its effectiveness can be slightly modulated by the uncertainty measure used.

5.4 F1 and Recall Dynamics in KP AVG Strategies: Actual vs. Reverse

KP AVG Reverse consistently achieves higher F1 scores than KP AVG Actual across cycles, as shown in Tables 3, 4 and 5 and Figure 7 for MTP uncertainty measure. This suggests that selecting documents with lower keyphrase uncertainty (as done in Reverse) helps the model learn to reduce false positives, improving precision and overall F1.

Although KP AVG Actual was expected to improve recall by selecting documents with higher keyphrase uncertainty, the observed recall improvements are marginal and inconsistent compared to the Reverse strategy, as shown in Figure 8. This indicates that the hypothesized advantage of KP AVG Actual in exposing the model to uncertain keyphrase boundaries may not be strong enough to translate into significantly better recall.

5.5 Comparing Early-Stage Improvements: Reverse vs. Actual Order

We hypothesized that Reverse order would yield quicker improvements in the early stages (θ□ to θ□) and that Actual order would show more sustained improvements later (θ□ to θ□ or θ□). This is why we experimented with both, despite Actual order being the traditional choice in Active Learning. However, the observed pattern did not hold across most strategies, possibly due to the limited dataset. While Reverse order shows quicker gains initially, Actual order did not consistently lead to stronger long-term improvements as expected.

5.6 Active Learning vs. Random Selection: Impact on Recall and Model Stability

The results from our experiments (Tables 3 and 5) show that while both the Active Learning (KPSUM Actual) and Random selection strategies yield improvements in performance, the key difference lies in the stability of these improvements. In our experiments, KPSUM Actual demonstrates more reliable gains, especially after the initial learning cycles. Once the saturation point is reached, performance metrics stabilize and show less fluctuation, indicating consistent improvement over time. This is further illustrated in Figures 9 and 10, where the F1 and Recall scores for KPSUM Actual show a more consistent upward trend compared to Random selection.

On the other hand, Random selection, though it shows improvements initially, tends to exhibit greater volatility. After a certain point, the recall and F1 scores for Random selection can decline, highlighting the inefficiency of random document addition in refining the model's performance. This suggests that while Random selection may lead to some improvements, its learning trajectory is less predictable, and its performance may degrade once the model has been exposed to a certain amount of data. Figures 9 and 10 also demonstrate this fluctuation in Random selection, contrasting it with the more stable progression seen in KPSUM Actual.

These results align with findings from 2021 [Ref 25], which also observed that Active Learning strategies outperform Random selection in terms of stability and more consistent improvements over time. Our results further underscore the value of a structured, uncertainty-based approach for document selection, such as KPSUM Actual, which offers more reliable and stable performance gains in comparison to Random selection.

5.7 F1 scores and weighted F1 scores

F measures are typically used in machine learning, information retrieval, and NLP tasks. Although in our use case, recall is much more important than F measure, we still keep F measure in both original values and weighted values to provide a more balanced and comprehensive measure for model performance. From the tables 7, 8, and 9, we noticed

weighted F1 scores when Beta equals to 5 and 10 show similar value and trend. Weighted F1 values have shown approximately 0.03 to 0.05 lower than the original F1 values across different strategies. Moreover, weighted F1 values peaked at θ□ constantly across most strategies, including random sampling. The original F1 value peaked inconsistently, including θ□, θ□, and θ□. Overall, weighted F1 values provide more comprehensive perspectives on the model performance and are important complementary measures to original F1 values and recalls.

5.8 Exploratory Experiments and Alternate Strategies

We attempted topic clustering of the documents to explore whether adding a document from a particular cluster would improve the model's performance on other documents within the same topic. However, this approach did not yield significant results. We think the limited number of documents might make it difficult to form reliable and meaningful clusters. With a larger dataset, topic clustering could be more feasible to test whether adding one document from a particular cluster could impact performance across different clusters.

Additionally, we considered the use of a weighted mean as an aggregation strategy. Conceptually, this would serve as a compromise between DocAvg (which averages uncertainty across all tokens) and KPAvg (which focuses only on keyphrases). By adjusting the weight ratio between keyphrase and non-keyphrase tokens, one can control the emphasis placed on keyphrase uncertainty. Though promising, we were unable to experiment with this due to time constraints.

It is important to note that the observed differences across methods were relatively small, likely due to our limited Active Learning data pool. With a larger pool, more pronounced and reliable trends might emerge.

5.9 How will the results be used?

We started the investigation aiming to facilitate the long-term maintenance of the CDSS ontology. The results presented in this paper show promise; however, they may not be ideal yet. Considering the complexity of our task: 1) not a binary prediction, 2) the three HDEs could not reach 100% consensus even after discussions, we are very pleased with the initial exploration and the results. Realistically, workflow automation is necessary to make the long term ontology maintenance feasible; however, considering the quality requirements of ontology, the current technology has not reached the level that can be relied on solely in operation. This can also be demonstrated by the work using LLMs to extract semantics [Ref 15], which demonstrated the exploration of using most advanced technology to achieve similar goals, however, with comparable results with ours. This indicates the task is indeed challenging and there is plenty of space for improvement. Therefore, putting HDE in the loop for our task is necessary in the foreseeable future. We think that we are on the right track to use active learning to facilitate the task. Meanwhile, continuing to explore ways to improve the model performance is a needed future priority.

5.10 Limitations and Future Work

Our study has several limitations: 1) the performance of the current active learning pipeline can be improved further. The current performance, despite being promising considering the complexity of the task, can be limited in real-world use. In the future, we should work on improving the performance of the pipeline. 2) In this study, we have a limited number of ground truth documents, and the task is complex and very unique. In the future, with a larger number of ground truth documents, others may obtain different results. However, we believe the approach described in this paper can still be referenced. 3) In this study, we explored basic aggregation strategies for estimating document-level uncertainty, primarily focusing on sum-based (e.g., KPSum, DocSum) and average-based (e.g., KPAvg, DocAvg) methods. The future work can include richer explorations by implementing advanced mathematical compositions, such as exponential or logarithmic calculations. In addition, future work could explore alternative aggregation techniques such as trimmed mean, which reduces the influence of extreme uncertainty scores, or weighted mean, where keyphrase scores are weighted based on contextual importance. Techniques like median, mode, or box-plot-based selection (e.g., interquartile range filtering) may also provide more robust document representations, especially in skewed distributions. In addition, the real impact of active learning in HDE annotation tasks in ontology construction and maintenance context can also be measured based on work saved over sampling algorithms [Ref 26, 27, 28, 29].

Section 6: CONCLUSION

In this study, we proposed a generalizable Active Learning framework for identifying candidate keyphrases for a CDSS ontology using a BiLSTM-CRF model, by incorporating HDE feedback. A key contribution of our work is the design and evaluation of novel document-level uncertainty aggregation strategies (KPSum, KPAvg, DocSum, and DocAvg), which enable more effective, transparent, and reproducible document selection for annotation. Based on our experimental results, KPSum (Actual) emerged as the most effective strategy for improving keyphrase identification performance in our setting. Meanwhile, the other aggregation methods presented in this manuscript provide a foundation for peers to compare and select the optimal approach and measure for their own tasks and datasets. Through systematic comparison of various Active Learning strategies and uncertainty measures, this study not only improves keyphrase identification performance but also offers practical benchmarks for future research in similar domains. While our experiments centered on keyphrase identification, the proposed framework is domain-agnostic and can be adapted to other sequence labeling tasks. Overall, this work contributes to the underexplored application of Active Learning in deep learning-based keyphrase identification, providing a scalable and adaptive foundation for efficient HDE annotation in NLP.

Acknowledgement

This study is made possible by funding from the National Institute of General Medical Sciences (R01GM138589) at the National Institutes of Health and partially through P20GM121342 of NIGMS and partially through # OIA-2242812 of National Science Foundation.

REFERENCES:

1. Greenes, R. and G.D. Fiol, Clinical Decision Support and Beyond: Progress and Opportunities in Knowledge-Enhanced Health and Healthcare. 3rd Edition ed. 2023, Cambridge, MA: Elsevier: Academic Press.[a]
2. Davies, J., D. Fensel, and F.V. Harmelen, Towards the semantic web: Ontology-driven knowledge management. 2003, Chichester: John Wiley & Sons.[b]
3. Supekar, K., et al., Knowledge zone: A public respiratory of peer-reviewed biomedical ontologies in MEDINFO 2007, K.e.a. Kuhn, Editor. 2007, IOS Press: Brisbane, Australia. p. 812-816. Rector, A.L. Modularisation of domain ontologies implemented in description logics and related formalisms including OWL. in K-CAP'03. 2003. Sanibel Island, Florida, USA.[c]
4. Rector AL. Modularisation of domain ontologies implemented in description logics and related formalisms including OWL. Proceedings of the 2nd international conference on Knowledge capture. 121-128. K-CAP'03; 2003 October 23–25, 2003; Sanibel Island, Florida, USA.[x]
5. Liu, M., Tu, Z., Zhang, T. *et al.* LTP: A New Active Learning Strategy for CRF-Based Named Entity Recognition. *Neural Process Lett* 54, 2433–2454 (2022). https://doi.org/10.1007/s11063-021-10737-x [MTP, TE, MARGIN][d]
6. Rose, S.J., Engel, D.W., Cramer, N., & Cowley, W. (2010). 1 Automatic keyword extraction from individual documents. (TF IDF for Keyword retrieval) [e]
7. Lafferty, J.D., McCallum, A., & Pereira, F. (2001). Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. *International Conference on Machine Learning*. (CRF for sequence tagging) [f]
8. Huang, Z., Xu, W., & Yu, K. (2015). Bidirectional LSTM-CRF Models for Sequence Tagging. *ArXiv, abs/1508.01991*.(BiLSTM CRF Model) [g]
9. Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *North American Chapter of the Association for Computational Linguistics*.(BERT) [h]
10. Goli R, Hubig N, Min H, Gong Y, Sittig D, Robinson D, et al. Keyphrase Identification Using Minimal Labeled Data with Hierarchical Context and Transfer Learning. BMC Medical Informatics and Decision Making. Under review. MedRxiv Preprint. 2023. https://medrxiv.org/cgi/content/short/2023.01.26.23285060v2[i]
11. Monarch, R.M., Human-in-the-Loop Machine Learning: Active learning and annotation for human-centered AI. 2021, Shelter Island, NY: Manning Publications.[j]
12. Settles, B. (2009). Active Learning Literature Survey. [k]

13. Yanyao Shen, Hyokun Yun, Zachary Lipton, Yakov Kronrod, and Animashree Anandkumar. Deep active learning for named entity recognition. In Proceedings of the 2nd Workshop on Representation Learning for NLP, pp. 252–256, Vancouver, Canada, August 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-2630. URL https://www.aclweb.org/anthology/ W17-2630. [l]
14. Aditya Siddhant and Zachary C. Lipton. 2018. Deep Bayesian active learning for natural language processing: Results of a large-scale empirical study. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2904–2909, Brussels, Belgium. Association for Computational Linguistics. [m]
15. Caufield J, Hegde H, Emonet V, Harris N, Joachimiak M, Matentzoglu N. Structured prompt interrogation and recursive extraction of semantics (SPIRES): A method for populating knowledge bases using zero-shot learning. Bioinformatics. 2024;40(3).[y]
16. Puria Radmard, Yassir Fathullah, and Aldo Lipani. 2021. Subsequence Based Deep Active Learning for Named Entity Recognition. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4310–4321, Online. Association for Computational Linguistics.[n].
17. Katerina Margatina, Giorgos Vernikos, Loïc Barrault, and Nikolaos Aletras. 2021. Active Learning by Acquiring Contrastive Examples. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 650–663, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics. [o]
18. S. -J. Huang, R. Jin and Z. -H. Zhou, "Active Learning by Querying Informative and Representative Examples," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 36, no. 10, pp. 1936-1949, Oct. 2014, doi: 10.1109/TPAMI.2014.2307881.keywords: {Measurement uncertainty;Uncertainty;Algorithm design and analysis;Correlation;Labeling;Clustering algorithms;Kernel;Active learning;learning with unlabeled data;multi-label learning;informativeness;representativeness}.[p]
19. Jingbo Zhu, Huizhen Wang, Tianshun Yao, and Benjamin K Tsou. 2008. Active Learning with Sampling by Uncertainty and Density for Word Sense Disambiguation and Text Classification. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 1137–1144, Manchester, UK. Coling 2008 Organizing Committee. [q]

20. Dara Bahri, Heinrich Jiang, Tal Schuster, and Afshin Rostamizadeh. 2022. Is margin all you need? An extensive empirical study of active learning on tabular data, arxiv. DOI: https://arxiv.org/abs/2210.03822[(Margin uncertainty boundary)][r]
21. Neha Gupta, Hari Narasimhan, Ankit Singh Rawat, Wittawat Jitkrittum, Aditya Menon, and Sanjiv Kumar. 2024. Language Model Cascades: Token-Level Uncertainty And Beyond, arxiv. DOI: https://doi.org/10.48550/arXiv.2404.10136 (KPSUM, KP AVg Summing)[s]
22. Jing X, Goli R, Komatineni K, Alluri S, Hubig N, Min H, et al. Active Learning Pipeline to Identify Candidate Terms for a CDSS Ontology. Stud Health Technol Inform. 2024;316:1338-42. DOI: 10.3233/shti240660 [v]
23. Jing X, Indani A, Hubig NC, Min H, Gong Y, Cimino JJ, et al. A systematic approach to configuring MetaMap for optimal performance. Methods Inf Med. 2022. DOI: 10.1055/a-1862-0421[w]
24. Ishibashi, H., & Hino, H. (2020). Stopping criterion for active learning based on deterministic generalization bounds. *ArXiv, abs/2005.07402*.[u]
25. Mosqueira-Rey, E., Alonso-Ríos, D. & Baamonde-Lozano, A. Integrating Iterative Machine Teaching and Active Learning into the Machine Learning Loop. *Procedia Comput. Sci.* 192, 553–562 (2021) (Random)[t].
26. Wallace BC, Small K, Brodley CE, Trikalinos TA. Active learning for biomedical citation screening. Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining; Washington, DC, USA: Association for Computing Machinery; 2010. p. 173–82.[z]
27. Wallace BC, Small K, Brodley CE, Trikalinos TA. Who Should Label What? Instance Allocation in Multiple Expert Active Learning. Proceedings of the 2011 SIAM International Conference on Data Mining (SDM). p. 176-87.[zz]
28. Byron C. Wallace, Kevin Small, Carla E. Brodley, Joseph Lau, Trikalinos TA, editors. Modeling Annotation Time to Reduce Workload in Comparative Effectiveness Reviews. Proceedings of the 1st ACM International Health Informatics Symposium; 2010 Nov 11-12, 2010; Arlington, VA.[zzz]
29. Byron C. W, Trikalinos TA, Lau J, Brodley C, Schmid CH. Semi-automated screening of biomedical citations for systematic reviews. BMC Bioinformatics. 2010;11:55-67.[zzzz]
30. Scikit-Learn. fbeta_score 2025 https://scikit-learn.org/stable/modules/generated/sklearn.metrics.fbeta_score.html.