Team IDACCS at TREC 2025: RAG and RAGTIME Tracks

John M. Conroy[1], Mike Green[3], Neil P. Molino[1], Yue “Ray” Wang[2] and Julia S. Yang[1]

[1] IDA Center for Computing Sciences, {conroy,npmolin,jsyang}@super.org
[2] University of North Carolina at Chapel Hill, wangyue@unc.edu
[3] Department of Defense, USA; magree22@ncsu.edu

Abstract

This paper gives an overview of team IDA/CCS’s submissions to the 2025 TREC RAG and RAGTIME tracks. Our approach builds on our 2024 RAG (team LAS) and NeuCLIR (team IDA/CCS) approaches. We started with the 2024 NeuCLIR task, fine-tuning it for the NeuCLIR pilot data. We then adapted this approach for both the RAGTIME and RAG generation task submissions. As the 2025 RAGTIME task is multilingual, instead of cross-lingual like 2024, it was natural to look at stratified retrieval and compare it to a multilingual ranking using the NeuCLIR pilot data. We found that stratified query retrieval with reranking, adapted from our RAG 2024 work, was particularly helpful for generating reports within 2K and 10K character limits. In addition, we show work on improving extraction, using `occams` and attribution. Finally, we include a detailed meta-analysis of the automatic and semi-automatic metrics.

Keywords: cross-lingual retrieval; cross-lingual text summarization; attribution; retrieval reranking; retrieval-augmented generation

1 Introduction

This paper focuses on efforts to improve multi-lingual retrieval-augmented generation (RAG). It also focuses on the problem formulation and evaluation methods used at the Text Retrieval Evaluation Conference (TREC). The data used in this work were from the RAG TREC Instrument for Multilingual Evaluation (RAGTIME) track. We will give a brief background of the task here, but for a more in-depth overview, see [10].

This work incorporates research from several previous TREC submissions by the authors [1][2][7][13]. This includes research in the multilingual generation, evaluation, and attribution.

The RAGTIME track grew from TREC NeuCLIR 2024, the multilingual retrieval task of 2023 and 2024. An overview of this task may be found here [3].

We made a significant effort to understand the manual, automatic, and semi-automatic evaluation methods used for RAGTIME. Their overall framework is based on a query-answer pair formulation of an information unit or “nugget,” as well as an evaluation of the attribution of sentences in the generated report (summary). Their evaluation framework is called Automated Report Generation Under Evaluation or (ARGUE) and was introduced in a paper at SIGIR [5].

1.1 Data

We now give a brief overview of the NeuCLIR 2024 and RAGTIME 2025 data. TREC NeuCLIR 2024 and TREC RAGTIME 2025 utilized data from the Common Crawl, with Persian, Russian, and Chinese as the languages for 2024. For 2025, Persian was replaced with Arabic, and English source documents were added.

TREC RAGTIME 2025 used a test collection consisting of more than 4 million news articles in four languages (Arabic, Chinese, English, and Russian). A TREC RAGTIME 2025 query was similar in style to a TREC RAG 2025 query, with the addition of personalization. A sample of a TREC RAGTIME 2025 query is provided below:

> *I am the program chair for an artist co-op's summer workshop series. We have invited an expert glassblower from Syria to give a talk, and I want to have some background knowledge so that I can introduce the speaker and provide a high-level overview of the subject before their talk and demonstration, which will be more focused on technique. My introduction should cover some of the important features of Syrian glassblowing, the history of glassblowing in Syria, and why UNESCO sees it as an urgent matter to protect the traditional craft.*

See [10] for more details about the problem setup and data sets for the TREC RAG and RAGTIME tracks for 2025.

2 Systems and Methods

First, we employed hybrid techniques that integrate traditional and mathematically well-founded extractive text summarization, `occams`, [11], with the fluency and readability of an LLM, which paraphrases the sometimes-disjointed text from an extractive system. This has many advantages. The extractive summarizer is a form of context compression. This saves on costs because the LLM does not need to process as many tokens, yet is still exposed to the most salient information. It has also been shown to improve the quality of the generated response. In this work, it also helped control the length of the generated summary.

2.1 Hybrid Summaries

The hybrid summaries followed a pipeline starting with retrieval and ending with attribution. The overall flow is:

1. We used PLAID-X, a multilingual retrieval algorithm provided by the RAGTIME organizers, to retrieve the top 30 documents in each language using the background and problem statement as a query.

2. We reranked the documents to get the top 10 using `mxbai-rerank-large-v1` on 10 sentence chunks with an overlap of 5, using a query generated by GPT-4o based on the title, background, and problem statement.

3. An `occams` extractive summary of length twice the target length was produced.

   (a) For a total target length of 10000, the target length was set to be 2500 per language (Arabic, Chinese, English, and Russian) as the generation was done per language.

   (b) For a target length of 2000, the top 30 translated documents from each language were pooled, and the top 10 were used to generate the extractive summary.

4. GPT-4.1 was prompted to do one of the following:
   (a) Form "nuggets" not to exceed the target length, used to generate the report.
   (b) Compose a "fluent" summary from the extracts.
   (c) Form an extract from the `occams` extract.
5. Attribution was done using our "blame" method, which was an outgrowth of research done in 2023-2024 at the Summer Conference for Applied Data Sciences at the Laboratory for Analytic Sciences [1]. As documented in the TREC notebook [2], we found that the `sentence-t5-base` model performed as well as the `sentence-t5-large` model for attribution in the cross-lingual NeuCLIR pilot sample data [6].

We tested adding the nugget variation to see if a small prompt change would make the hybrid summaries more concise and improve coverage. In testing on the NeuCLIR pilot track, the approach tended to produce more concise statements, but did not measurably improve coverage (nugget recall). The extraction option was added to ensure that one submission would be a direct extract and trivial to attribute. The LLM, in this case GPT-4o or GPT-4.1, is used to refine the `occams` extractive summary. As the RAGTIME evaluation does not include fluency, we surmised that the submission should have excellent attribution with some loss of coverage relative to the nuggets, as extracted sentences will be longer on average than the shorter nuggets produced by the LLM from the `occams` summary.

Because the cost of GPT-4.1 is higher than that of GPT-4o, we did the bulk of our testing using GPT-4o. For the submissions, we compared the lengths of the summaries generated by these two models and found that GPT-4.1 produced more extended summaries, while GPT-4o tended to undergenerate. The abstractive summaries were tested but not submitted, as they were too short: when a target length of 2000 was requested, the mean length was approximately 1300, and for 10000, it was about 5500. These submissions would likely have relatively low nugget recall, and we opted for other alternatives to the hybrid and nugget approaches for our submissions to RAGTIME.

2.1.1 Using Organizers' Search Service

To help the RAGTIME track participants, the track organizers provided a document search and retrieval service for the track's multilingual corpus. Details can be found on the track website `https://trec-ragtime.github.io/search_api.html`. It uses PLAID-X, a very effective cross-lingual information retrieval algorithm [12].

This was immensely helpful, and we used it almost exclusively for our document search and for retrieving the documents' text. It supported both the original languages and the machine-translated documents. We had systems that used the original, the translations, and both.

3 Results

In the TREC RAGTIME 2025 track, we submitted a hybrid summarization pipeline. As the TREC RAG track supports a similar format, we also submitted five approaches to the augmented generation (AG) task for their 400-word summary task.

For the AG task for RAG, the organizers provided the top 100 documents. With about 60 lines of code, we adapted the hybrid pipeline from the RAGTIME task for this task. Here we submitted the nugget and hybrid approaches for GPT-4o and GPT-4.1, as well as an abstractive approach using GPT-4.1. As with the RAGTIME experience, GPT-4o tended to undergenerate, producing well below the requested 400 words, which is unsurprising, as 400 words exceeds the 2000-character limit of RAGTIME.

3.1 Stratified Sampling in Cross-Lingual Retrieval

One of the major outcomes of our work on improving the RAGTIME pipeline was length control and improved nugget recall. Some main findings are

1. Using a stratified retrieval, i.e., getting the top n documents from each language, gave superior coverage (nugget recall) than a single multilingual (i.e., searching English, Russian, Chinese, and Arabic simultaneously) information retrieval query of the top $4n$ documents.
2. The hybrid approach not only allows for stronger attribution, but it also makes the LLM produce summaries closer to the 2000 and 10000 target lengths or the 400 words in the RAG track.
3. GPT-4.1 was better at achieving the longer target lengths for the RAGTIME track than GPT-4o.

3.2 On NeuCLIR Pilot Generation Task Results 2024

The RAGTIME evaluation includes five metrics, computed across 20 topics from the 2024 pilot generation task. The human evaluation framework is based on a position paper from SIGIR, which proposed a question-and-answer “nugget "- based evaluation. A software package called `argue_eval` implements automatic and semi-automatic variants of the five ARGUE scores. Figure 1 shows how the systems performed in the Russian source document task. The queries and summaries are in English. The results for Persian and Chinese are similar.

In Figure 1, the IDA_CCS_hybrid, an `occams`/GPT-4o submission, is second in the argue_score, due in part to its accuracy in attribution. The number one submission by Eugene Yang, JHU’s HLT/COE, a nugget extraction system, had a stronger nugget recall. Both systems performed well at attribution, which may in part be due to their approaches of nuggetizing or paraphrasing the extracted text. Note that attribution is trivial if a summary is extractive and, as the recent arXiv paper [1] demonstrates, it is easier to attribute sentences in a summary when a hybrid summary is generated than when an abstractive summary is generated, and semantic similarity via a sentence embedding does better than a natural language inference model. The ARGUE score depends on both the nugget recall and attribution, so finding the correct information and attributing it are both critical. Understanding the metrics is key to improving systems.

A meta-evaluation (an evaluation of the automatic metrics) comparing the semi-automatic `argue_eval` metrics to the human ARGUE scores was conducted for each pair of manual and semi-automatic metrics in each language, yielding 15 analyses. The evaluation is *semi*-automatic in that the LLM is used to judge if a human-derived nugget is contained in a summary. (The package `argue_eval`, like `open-rag-eval`, can also automatically generate nuggets, but this was not part of our study.) In each analysis, we compute the Kendall tau correlation as well as an empirical power analysis, where we compute the confidence level (fraction of time we do not reject the null when the manual statistic does not reject the null hypothesis) and power (fraction of the time that we reject the null hypothesis when the manual statistic does so). A combined measure of accuracy is reported, representing the probability that the approximate statistic’s outcome aligns with the computed statistic based on manual evaluation.

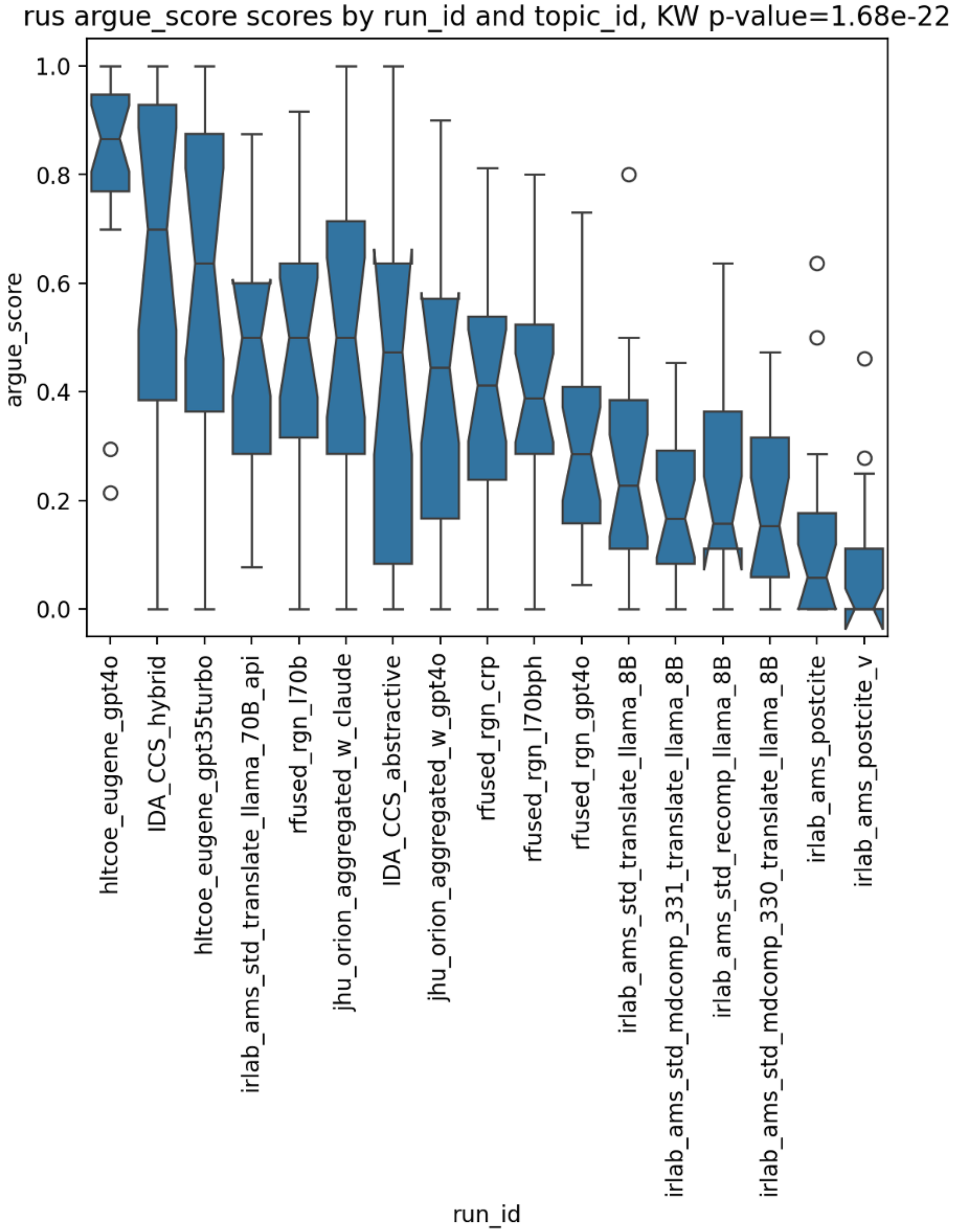

Figure 1: Box Plot of results from the NeuCLIR Pilot Generation Task

For brevity, we present the results for Russian and limit the comparison to the predicted `nugget_recall` and the `argue_score`, which depends on recall and the accuracy of the attributions. The results for Persian and Chinese were somewhat weaker. In figure 2, we observe that the semi-automatic nugget recall score best predicts the human-annotated nugget recall score, achieving an accuracy of approximately 81%. We note that the F1 score, which is the harmonic mean of the `nugget_recall` and `segment_precision`, is proposed to approximate the `argue_score`. For the Russian dataset, both the Kendall tau correlation coefficient and accuracy are lower than segment precision, which is the estimated statistic for sentence support.

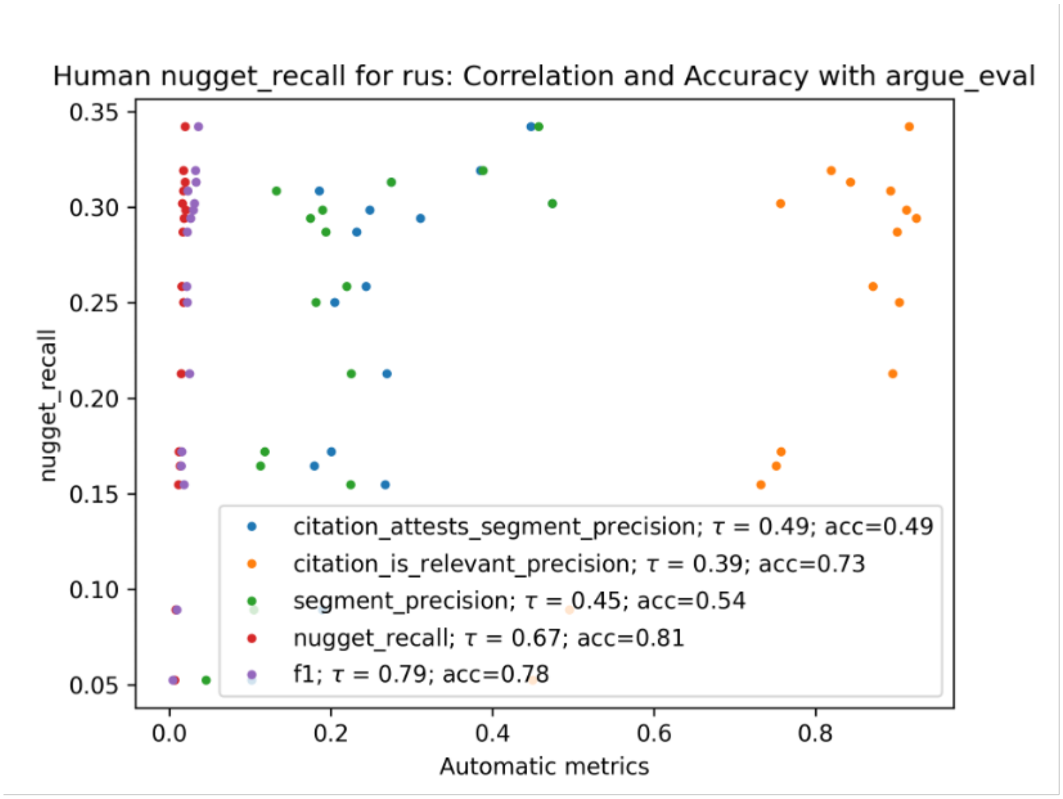

Figure 2: Meta Evaluation of the semi-automatic metric (nugget recall) as well as `argue_eval` metrics vs. the manual nugget recall metric.

3.3 Internal and External Evaluation

3.3.1 Evaluation Strategy

We used several avenues for evaluating our cross-lingual RAG systems. The official RAGTIME dry run provided a practical data point, but we also wanted a more dynamic way to measure progress internally. To that end, we adopted `open-rag-eval`, the LLM-based evaluation methodology described earlier. In the following subsection, we benchmark its use as a proxy for the official RAGTIME metrics.

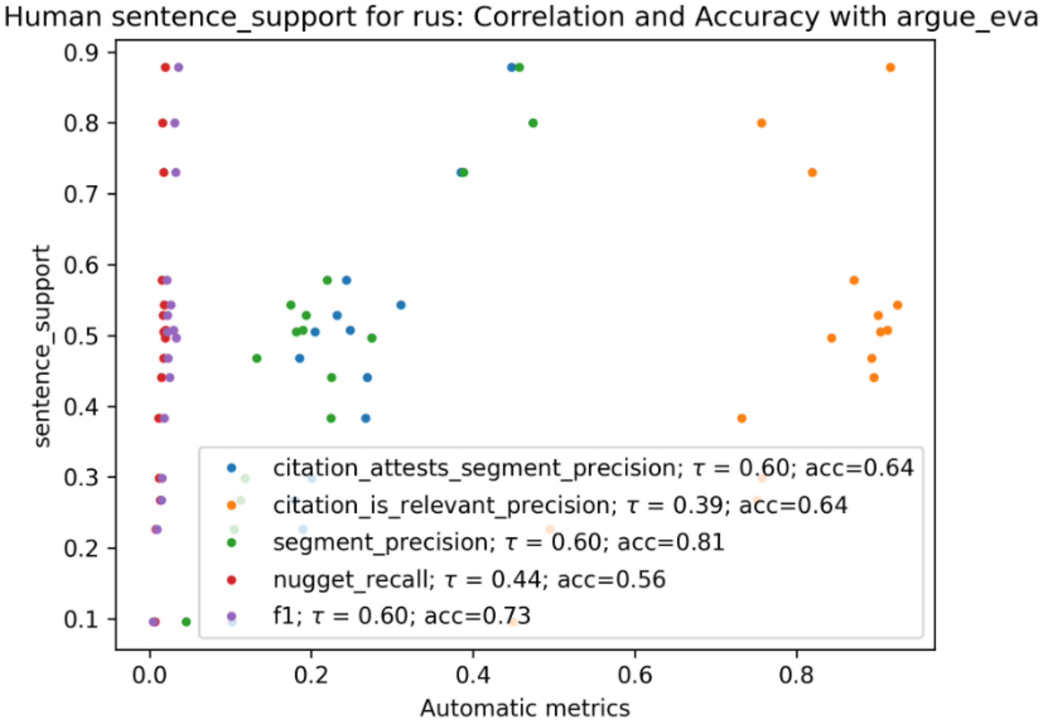

Figure 3: Meta Evaluation of the automatic metric (segment precision) as well as `argue_eval` metrics vs. the manual sentence support metric.

3.3.2 Benchmarking Open RAG Eval as a Proxy for ARGUE

The two primary metrics for RAGTIME will be sentence support and nugget coverage. They are both part of the ARGUE evaluation framework. To this end, we are interested in assessing how well Open RAG Eval can serve as a proxy for the actual scoring of our RAGTIME submissions.

We ran a simple correlation analysis on our scores on the systems that we submitted to the SCALE leaderboard (ARGUE-based) and the judgments made by Open RAG Eval (UMBRELA-based). The first plot below shows a heat map of correlations between the ARGUE and UMBRELA scores for all of the variants of our (hybrid and agentic) systems that we tested. The agentic runs are described in a separate notebook paper [8], which is the submission of Team LAS.

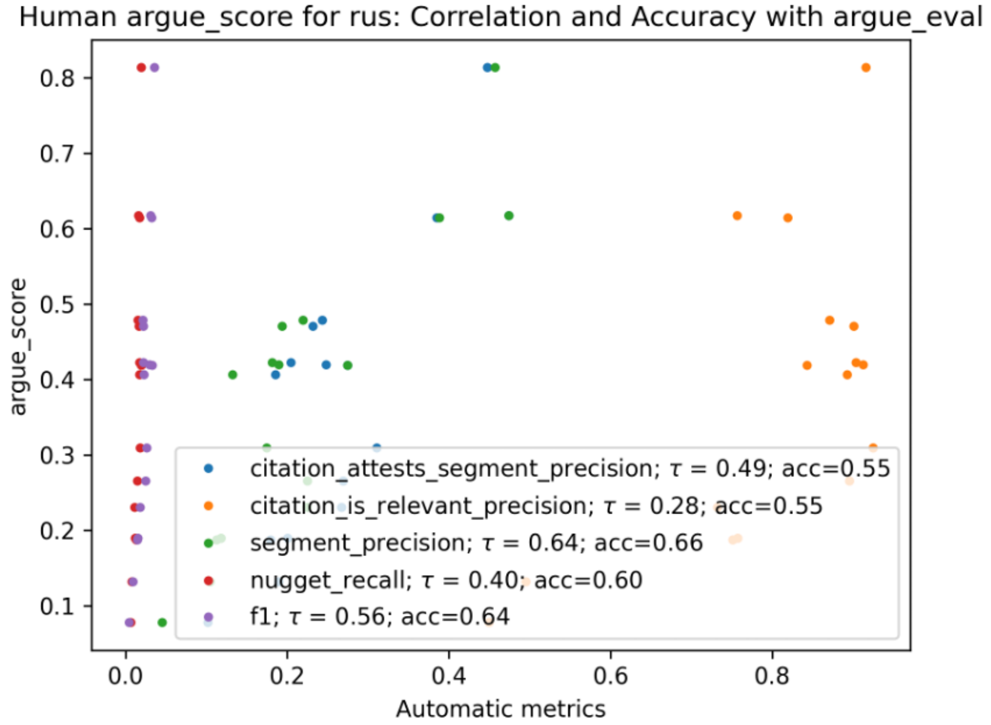

Figure 4: Meta Evaluation of the semi-automatic metric (F1) as well as `argue_eval` metrics vs. the manual `argue_score` metric.

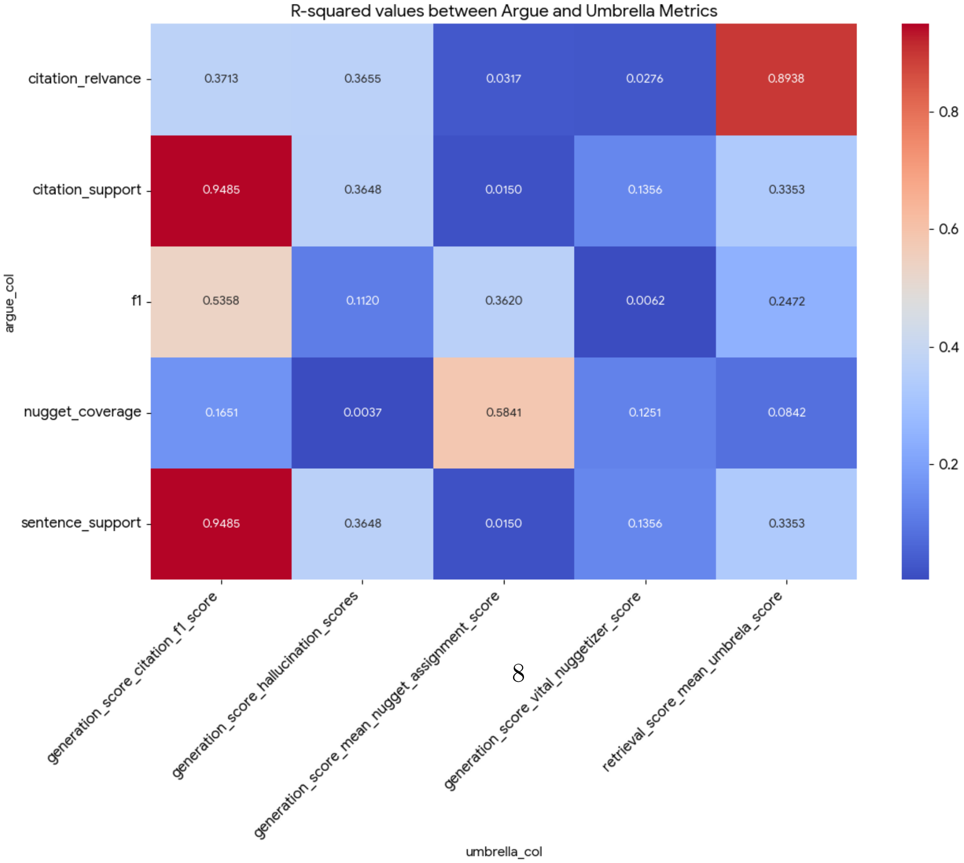

Figure 5: R-squared of UMBRELA metrics vs. ARGUE metrics

The heat map identifies which UMBRELA variables can serve as proxies for those variables. It is pretty clear in Figure 5 that ARGUE's sentence support metric correlates extremely well with UMBRELA's `generation_score_citation_f1_score`. The best one for nugget coverage is `generation_score_citation_f1_score`. Note that the values in the table are R-squared values and not correlation coefficients. A value of 0.49 in the table actually represents a correlation of 0.7.

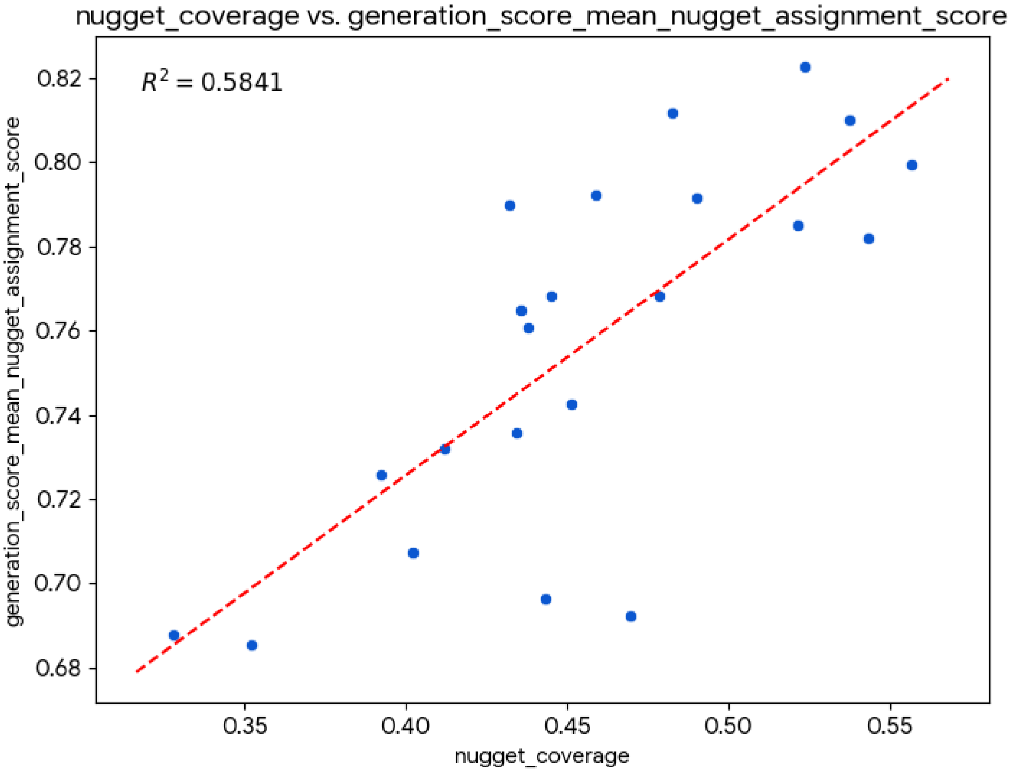

Figure 6: Correlation between ARGUE Nugget Coverage and UMBRELA Vital Nuggetizer Score

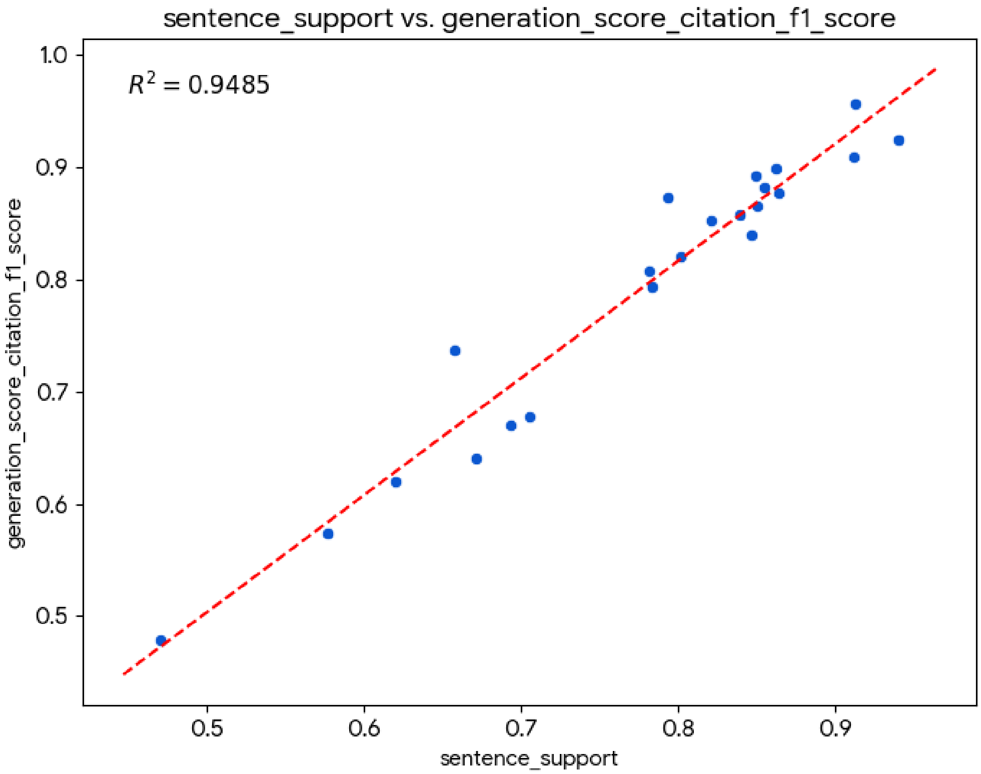

Figure 7: Correlation between ARGUE Sentence Support and UMBRELA Citation F1 Score

In figures 6 and 7, we can see the actual scatter plots of the values as well as the best-fit line. It is visually quite clear that the fit is much stronger for the attribution metrics than for the content.

Because the Open RAG Eval UMBRELA scoring is based on LLM judgments, it inherently exhibits some variability.

The plots below examine paired scores from two runs of UMRELLA/Open RAG Eval on a duplicate submission to gain insight into the intrinsic variability.

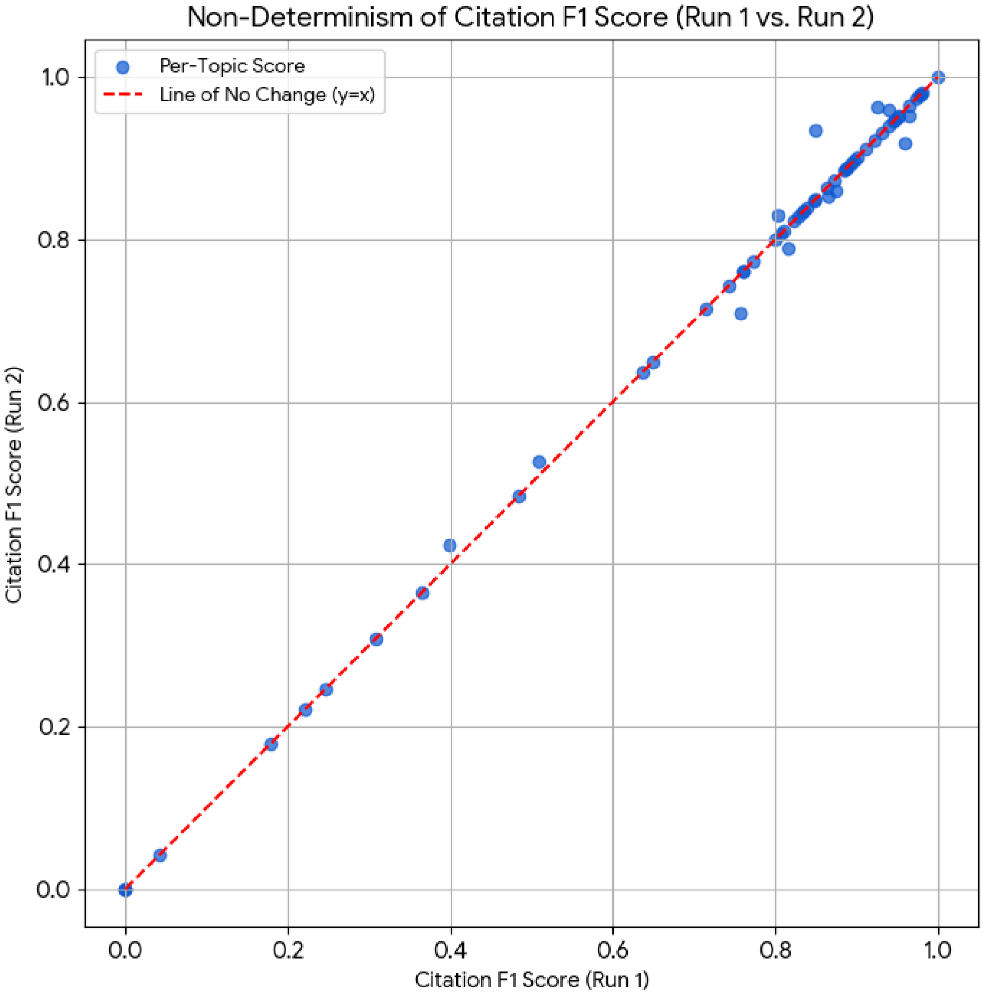

Figure 8: Attribution Scores Across Two Open RAG Eval Runs of the same Report Submission

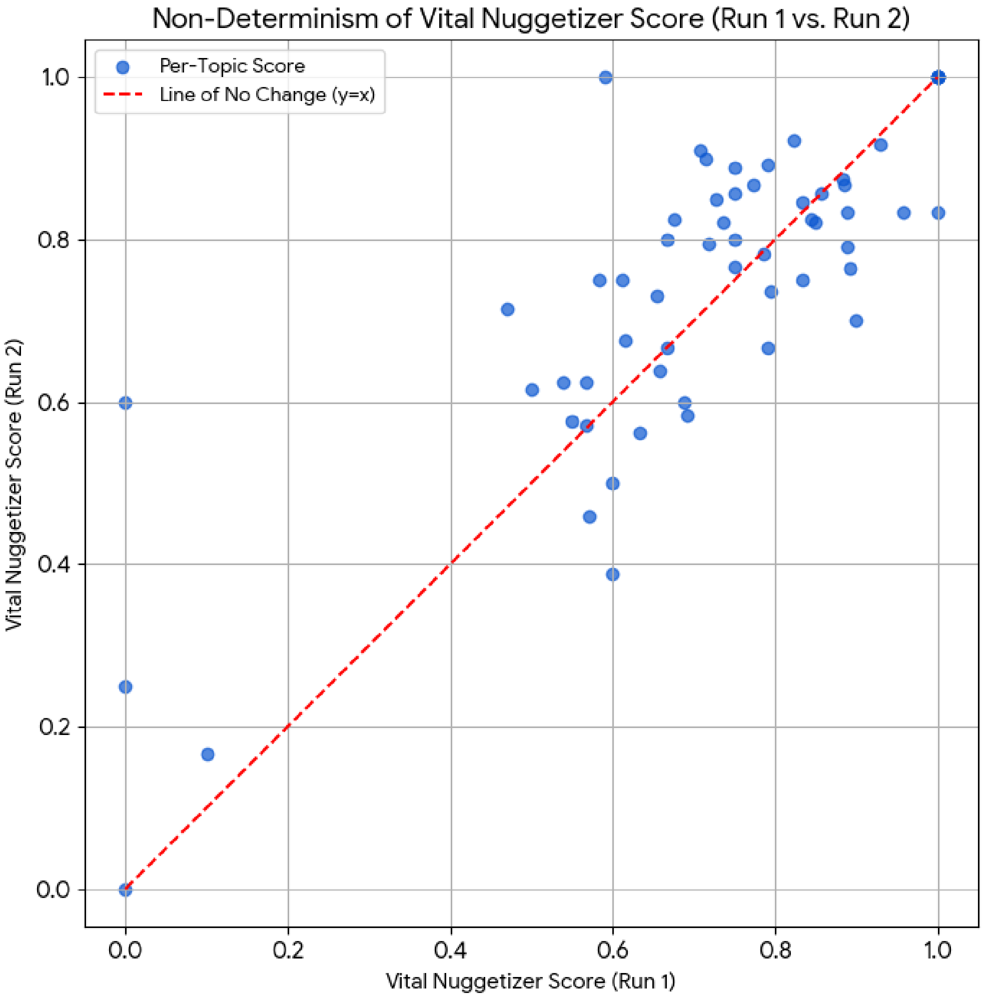

Figure 9: Content Scores Across Two Open RAG Eval Runs of the same Report Submission

Each dot represents a topic's score with the first run's score as the x coordinate and the second one as the y coordinate. If the scoring were perfectly deterministic, the plots would all lie along the diagonal line $y = x$. The citations are much closer to the diagonal line than the nuggets/content.

3.3.3 Content vs. Length

Another tricky facet of scoring content is the relationship with length. Clearly, the longer a piece of text, the more information it can contain. Our submissions have disparate nuggetizer scores, but how does length affect them? The graph below plots some of our systems.

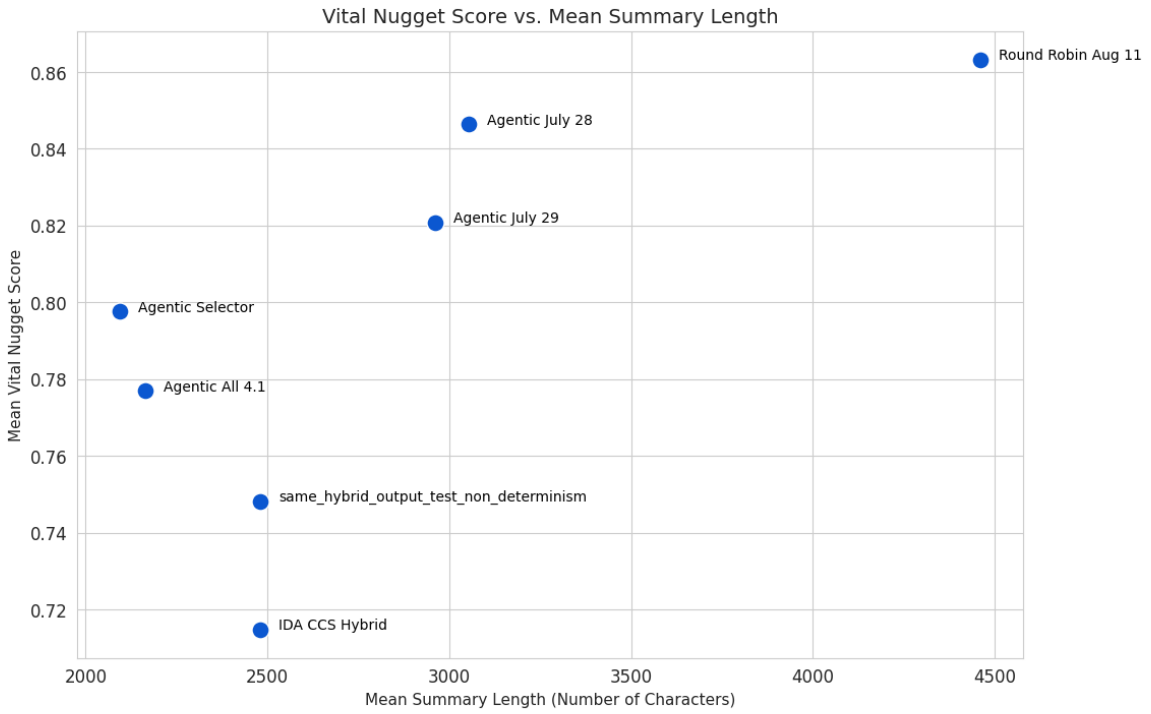

Figure 10: Length vs UMBRELA vital nugget score

As mentioned, the hybrid systems scored lower on the UMBRELA nugget score than the agentic ones. They are much better at controlling length, though. They use an extractive summarizer that guarantees its length will be below the target by solving a combinatorial optimization problem. It does call an LLM to paraphrase the extracted summary for readability. LLMs have difficulty controlling their output length, as can be seen in the high variance of agentic system response lengths in Figure 10.

3.3.4 RAGTIME Dry Run Performance

The RAGTIME track organizers provided the opportunity to participate in a mid-summer “dry run” submission. They released 25 dry run topics. The dry run was due on July 8, 2025. They released the results on August 11, 2025, a week before the final submissions were due. Only 18 of the 25 topics were judged. The results of our three submissions can be seen in figures 11 and 12. They were all based on our hybrid summarization-based system.

Our `moreblame/miniblame` based attribution performed well overall. Two of our three submissions were above the mean. One of them substantially so.

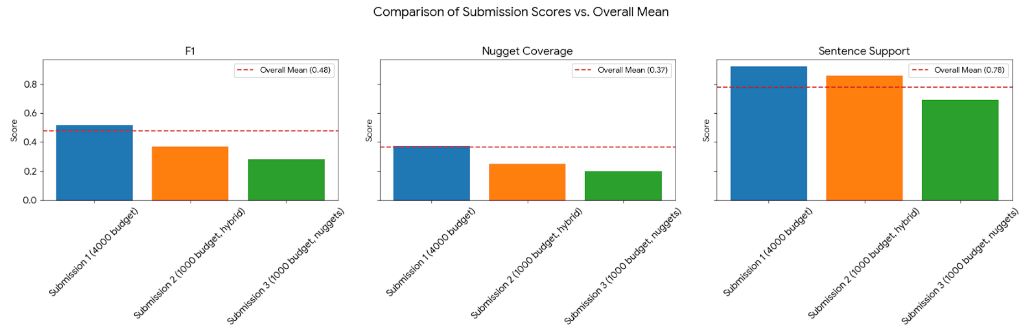

Figure 11: Our Hybrid Summarization Submissions

Our nugget coverage was less competitive. Only our most extended submission was above the mean on this metric. Our average score of 0.373 across all requests was slightly above the mean of 0.367.

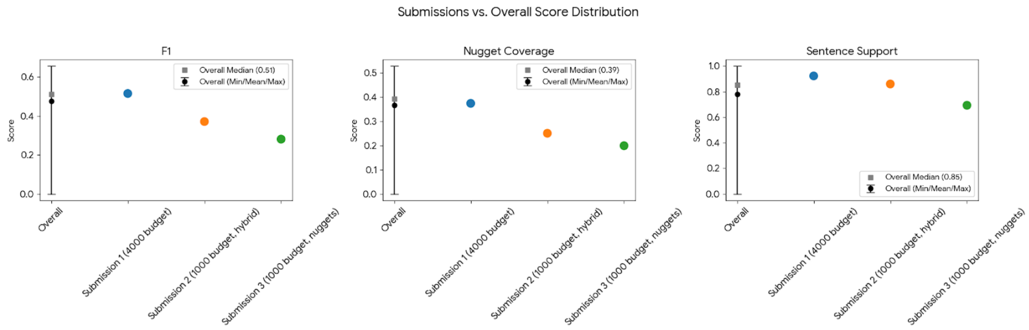

Figure 12: Our Hybrid Summarization Submissions

4 Discussion

Introducing multilingual corpora into RAG and RAG evaluation is an important objective. NIST and the TREC RAGTIME track provide an excellent controlled setting to make progress in this arena.

Generally, our experiments with Open RAG Eval found that agentic methods produced better content (nugget scores), but hybrid systems excelled at attribution and length control. Interestingly, this was not the case on the leaderboard. The results of the actual final submission will be another critical data point. Future work could consider extractive summarization as a tool call available to AI agents. Similarly, there could be a paraphrasing agent. These ideas could create a best-of-both-worlds situation.

Evaluation of text-based systems remains challenging. Finding high-quality labeled data is problematic. Can we trust LLMs to judge the quality of responses? How well do their judgments correspond to those of the ultimate humans who have the information needed? We are grateful for the work that NIST has done over the years to enable a rigorous scientific process on these systems. It is now more critical than ever with the widespread adoption of AI.

5 Observations on the Evaluation Data

Both the RAG and RAGTIME tracks released evaluation for content and attribution. The former was based on human judgments of relevant nuggets. Both tracks' evaluation, as of the publication time, only included automatic evaluation of the attribution.

For the RAG track, the best submission of our hybrid summarization method scored below 0.4 for strict content evaluation and about 0.65 for the sub-narrative coverage. The top group of submissions, which includes our agentic submission [9] score in 0.45 to 0.5 range for strict and 0.75 to 0.8 for sub-narrative. So, the hybrid is weaker, but is a low-cost alternative to an agentic system. We estimate the cost for the hybrid summaries to be about \$0.01 and the agentic to be about \$0.50 based on our two team submissions, IDACCS (hybrid) and nscu-las (agentic).

The hybrid system was more competitive for the RAGTIME track. In content coverage for 2000 character summaries the best hybrid submission achieved about 0.37 vs. 0.43 for the best overall approach. But for the 10,000-character summaries the best and second best systems were the hybrid approach! We believe that the hybrid excelled at the longer summaries because it was much easier to control length with the hybrid approach since `occams` excels at generating extracts of a given length. The stronger performance on the RAGTIME vs RAG track we attribute to the fact that the parameters (other than the target output length) were tuned based on the neuCLIR 2024 pilot data, the precursor to the neuCLIR track.

The attribution results are a bit puzzling. While our hybrid system was among the top in precision and recall for the RAG track, achieving about 75% in both precision and recall, for the RAGTIME track the submission ranked near the bottom. It is unclear what happened here as the hybrid system used the same attribution code as in the neuCLIR pilot task from 2024 and the attribution was near the top. What is especially difficult to understand is that one of the "hybrid" submissions for the RAGTIME track was a hybrid `occams`/LLM extract, where attribution should be trivial. We will wait for the human evaluation to help understand this result further.

A `occams` Experiments

We investigate both an unsupervised and supervised variation of `occams`, as it is a key part of our pipeline. Since our approach uses `occams` for the extractive portion of the hybrid, an improved extractive summarizer would likely improve the overall generation.

A.1 Using SPLADE as a Tokenzier in `occams`

The unsupervised approach was an idea that grew out of discussions on how to use SPLADE to improve `occams` term coverage. SPLADE is a highly efficient method for leveraging the BERT family of models to expand both the query and the document tokens. SPLADE uses classical indexing, enabling it to achieve similar performance to neural dense retrieval with reduced computational and storage requirements. As an example, we note SPLADE performed well at RAG track 2024, comparable to using `t5-xxl` with exact nearest-neighbor search [7]. The idea was to use SPLADE to expand terms when computing a term-sentence matrix. This is easily achieved by just passing `occams` a SPLADE tokenizer. The code for the tokenizer was created via `Claude.ai.` The experiment was a good example of "fail quickly," as the table below shows the SPLADE tokenization, which has a free parameter to adjust the amount of term expansion. The upshot was that the SPLADE tokenizer was inferior to `nltk`'s stemming for unigrams and is far below the performance of `occams` default method of bigrams. The failure of this idea suggests that bigrams correlate more strongly with nuggets than with SPLADE-expanded tokens. Further analysis is needed to tease this information out of the experiments.

Method	ROUGE-1	ROUGE-2	ROUGE-3	ROUGE-4
occams(BIGRAM)	0.3387	0.1139	0.0551	0.0329
occams(UNIGRAM)	0.3152	0.0879	0.0414	0.0250
occams_splade(0.75)	0.2988	0.0890	0.0410	0.0236
occams_splade(1.25)	0.2945	0.0860	0.0383	0.0211
occams_splade(1.0)	0.2941	0.0813	0.0359	0.0194
occams_splade(0.5)	0.2911	0.0861	0.0385	0.0218
occams_splade(0.1)	0.2592	0.0642	0.0277	0.0165

Table 1: ROUGE scores for different expansion and summarization methods on CNN/DM.

A.2 Transfer learning with `occams`

Next, we turn to a supervised approach to improve `occams`. There is an example notebook in the repository that uses a classical feature analysis. The notebook requires a modest amount of training data of document-summary pairs. Three simple term features are used

1. First Occur: The log of the normalized position of the first occurrence of a term.
2. Fisher: The log of the p-value of a hypothesis test testing if the frequency of the term is the same as would be expected given the length of the text.
3. Log Probability: The probability that a term would be drawn at random from the text.

We trained a term-weight model using both the CNN/DM dataset and a collection of government reports, using the notebook `occams_supervised_term_weights`, available in the `occams` repository. This model was then tested on the human-written summaries from the neuCLIR Pilot. Initially, summaries generated by the term-weighting model trained on CNN/DM were relatively brief, averaging about 1,000 characters, despite the request for 2,000 characters. To address this, we developed a new model based on the government reports. We evaluated its performance on the neuCLIR report-generation data, which included English human summaries for the Persian, English, and Russian language subsets. The results are promising: the supervised model improved across all automatic metrics compared to the `occams` default, which is currently used in our RAGTIME code.

The resulting summaries were scored using the automatic evaluations of ROUGE-1, 2, 3, and 4, as well as BERTScore [14] and AutoACU (Automatic Atomic Content Unit) [4]. The mean scores of the supervised term-weighting model were compared with the default bigram method for `occams`, and, as shown in Tables 2 and 3, the supervised model outperforms the default, albeit by a small margin.

Table 2: ROUGE Scores by Summarizer

Summarizer ID	ROUGE-1 R	ROUGE-2 R	ROUGE-3 R
`occams_default`	0.447782	0.110598	0.027559
`occams_super_govreport_10K`	0.451110	0.119787	0.038891

Table 3: Additional Evaluation Metrics by Summarizer

Summarizer ID	ROUGE-4 R	BERTScore R	AutoACU R
`occams_default`	0.010438	0.583793	0.071100
`occams_super_govreport_10K`	0.017013	0.597130	0.113699

While the ROUGE gains are modest, they do show promise: the AutoACU, a nugget recall score, increased significantly. This indicates that a simple supervised approach generalizes to other data and can help improve the quality of extracts. The supervised approach uses three features, each based on the bigram distributions in the background texts, summaries, and the corpus of the text being summarized. As we had limited test data, we opted not to include it in our submissions; however, we will test the idea on the test data once it is released.

References

[1] Violet B, John M. Conroy, Sean Lynch, Danielle M, Neil P. Molino, Aaron Wiechmann, and Julia S. Yang. Where did you get that? Towards summarization attribution for analysts. *arXiv preprint arXiv:2511.08589*, 2025.

[2] John M. Conroy, Razieh Fathi, Daria Symlova, and Y. Kelly Wu. Cross-lingual hybrid and abstractive summarization with automatic attribution for the neuCLIR 2024 pilot report generation track. The Thirty-Third Text REtrieval Conference (TREC 2024), Gaithersburg, MD, USA, November 15-18, 2024, 2024.

[3] Dawn Lawrie, Sean MacAvaney, James Mayfield, Paul McNamee, Douglas W. Oard, Luca Soldaini, and Eugene Yang. Overview of the TREC 2024 NeuCLIR track. The Thirty-Third Text REtrieval Conference (TREC 2024), Gaithersburg, MD, USA, November 15-18, 2024, 2024.

[4] Yixin Liu, Alexander Fabbri, Yilun Zhao, Pengfei Liu, Shafiq Joty, Chien-Sheng Wu, Caiming Xiong, and Dragomir Radev. Towards interpretable and efficient automatic reference-based summarization evaluation. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 16360–16368, Singapore, December 2023. Association for Computational Linguistics.

[5] James Mayfield, Eugene Yang, Dawn Lawrie, Sean MacAvaney, Paul McNamee, Douglas W. Oard, Luca Soldaini, Ian Soboroff, Orion Weller, Efsun Kayi, Kate Sanders, Marc Mason, and Noah Hibbler. On the evaluation of machine-generated reports. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, page 1904–1915, New York, NY, USA, 2024. Association for Computing Machinery.

[6] Jianmo Ni, Gustavo Hernández Ábrego, Noah Constant, Ji Ma, Keith B. Hall, Daniel Cer, and Yinfei Yang. Sentence-T5: Scalable sentence encoders from pre-trained text-to-text models. *arXiv preprint arXiv:2108.08877*, 2021.

[7] Yue Wang, John M. Conroy, Neil Molino, Julia Yang, and Mike Green. Laboratory for Analytic Sciences in TREC 2024 retrieval augmented generation track. In *The Thirty-Third Text REtrieval Conference Proceedings (TREC 2024), Gaithersburg, MD, USA, November 15-18, 2024*, volume 1329 of *NIST Special Publication*. National Institute of Standards and Technology (NIST), 2024.

[8] Yue Wang, John M. Conroy, Neil Molino, Julia Yang, and Mike Green. Laboratory for Analytic Sciences in TREC 2025 RAG and RAGTIME tracks. In *Proceedings of the 34th Text Retrieval Conference (TREC 2025)*, Gaithersburg, Maryland, USA, 2025. National Institute of Standards and Technology (NIST). Notebook paper.

[9] Yue Wang, John M. Conroy, Neil Molino, Julia Yang, and Mike Green. Laboratory for analytic sciences in trec 2025 rag and ragtime tracks. In *TREC 2025 Notebook Proceedings*. NIST, 2025. University of North Carolina at Chapel Hill; IDA Center for Computing Sciences; Department of Defense, USA.

[10] Yue "Ray" Wang, Mike Green, John M. Conroy, Neil Molino, Julia Yang, Gideon M, and Joshua LG. SCADS team effort in two TREC 2025 retrieval-augmented generation tracks. In *The Fourth Summer Conference on Applied Data Science (SCADS 2025), Raleigh, NC, USA, June 2 - July 25, 2025*. NCSU Laboratory for Analytic Sciences (LAS), 2025.

[11] Clinton T. White, Neil P. Molino, Julia S. Yang, and John M. Conroy. `occams`: A text summarization package. *Analytics*, 2(3):546–559, 2023.

[12] Eugene Yang, Dawn Lawrie, James Mayfield, Douglas W. Oard, and Scott Miller. Translate-distill: Learning cross-language dense retrieval by translation and distillation. In *Proceedings of the 46th European Conference on Information Retrieval (ECIR)*, 2024.

[13] Julia Yang. Case study: Summary attribution. In *The Second Summer Conference on Applied Data Science (SCADS 2023), Raleigh, NC, USA, June 5 - July 28, 2023*. NCSU Laboratory for Analytic Sciences (LAS), 2023.

[14] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. BERTScore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.