pjmathematician at MultiClinSUM 2025: A Novel Automated Prompt Optimization Framework for Multilingual Clinical Summarization

Notebook for the BioASQ Lab, CLEF 2025

Poojan Vachharajani[1,*]

[1]Netaji Subhas University of Technology, New Delhi, India

Abstract

This paper describes the ‘pjmathematician‘ team’s submission to the MultiClinSUM 2025 shared task, focusing on multilingual summarization of clinical case reports in English, Spanish, French, and Portuguese. Our approach leverages fine-tuned Large Language Models (LLMs) from the Qwen family, adapted using Low-Rank Adaptation (LoRA). The core of our methodology is a novel, automated prompt optimization framework where a "judge" LLM iteratively refines the system prompt for a "worker" LLM to maximize summarization quality, measured by ROUGE scores. This process resulted in a highly-specific, extraction-focused prompt that instructs the model to mirror the source text’s terminology and structure with high fidelity. We submitted multiple runs using different model configurations, trained exclusively on the provided gold-standard dataset. Our results demonstrate the effectiveness of this automated prompt engineering strategy, achieving competitive scores across all four languages, with BERTScore F1 reaching up to 0.864 in English.

Keywords

Clinical Summarization, Large Language Models, Prompt Engineering, Automated Prompt Optimization, LoRA, MultiClinSUM

1. Introduction

The rapid accumulation of clinical content, such as electronic health records and case reports, presents a significant challenge for healthcare professionals who need to quickly synthesize key information. The MultiClinSum shared task at BioASQ 2025, organized by the Barcelona Supercomputing Center, directly addresses this by focusing on the automatic summarization of full clinical case reports in four languages: English, Spanish, French, and Portuguese [1]. The task aims to benchmark systems that generate concise summaries from complex clinical texts, supporting clinical decision-making and enhancing multilingual understanding. Evaluation is based on standard metrics including ROUGE and BERTScore, providing a crucial benchmark for NLP systems in this high-stakes domain.

This paper details the participation of the ‘pjmathematician‘ team in the MultiClinSUM task. Our approach centered on the use of fine-tuned Large Language Models (LLMs). Recognizing the profound impact of prompt design on LLM performance, our primary contribution is a novel, automated prompt optimization framework. This framework uses an LLM-to-LLM interaction to systematically discover a highly effective system prompt for the summarization task.

We employed a single-model strategy, training our systems on a combined dataset of all four languages. Our experiments, conducted exclusively on the provided gold-standard training data, demonstrate the viability of this approach. The resulting system achieves strong performance across all languages, underscoring the potential of advanced prompt engineering in specialized domains.

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain

*Corresponding author.

pjmathematician@gmail.com (P. Vachharajani)

2. Related Work

The automated summarization of clinical text is a long-standing challenge in natural language processing, driven by the need to condense vast amounts of clinical data from sources like electronic health records and medical literature to support clinical decision-making [15]. Early approaches were often extractive, relying on methods like TextRank. However, the advent of deep learning and transformer-based architectures has led to significant progress, with models like BERT and T5 being adapted for the clinical domain [5].

Recent research has heavily focused on the application of Large Language Models (LLMs) to this problem, demonstrating their potential to generate high-quality, coherent summaries [24, 18]. Studies have shown that with appropriate adaptation, such as fine-tuning, LLMs can produce summaries of clinical texts that are comparable or even superior to those written by medical experts [21]. This has been explored across a variety of clinical documents, including radiology reports, progress notes, and doctor-patient dialogues [12]. A significant portion of this research has been conducted on English-language data, often using datasets like MIMIC-IV [8]. While multilingual summarization is a recognized goal, as evidenced by the MultiClinSUM shared task itself [17], dedicated studies in this area remain less common.

A critical aspect of leveraging LLMs is prompt engineering, which has been shown to significantly influence model performance [27, 23]. The process of designing effective prompts is crucial in specialized domains like medicine, which has its own unique terminology and structure. Our work aligns with a growing body of research that seeks to move beyond manual prompt crafting towards more systematic and automated methods [7]. This includes techniques where an LLM itself is used to refine prompts. For instance, Pryzant et al. (2023) proposed a method using an LLM's feedback to generate "textual gradients" to iteratively improve a prompt [16]. Similarly, other optimization frameworks use an LLM to generate new prompts based on the performance of previous ones [10, 26, 6]. Our "judge-worker" framework is a novel contribution to this area of automated prompt optimization, specifically tailored for the complexities of multilingual clinical summarization.

Furthermore, our use of Low-Rank Adaptation (LoRA) for efficient fine-tuning is consistent with current best practices for adapting large models to specialized tasks. LoRA has been successfully applied in the clinical domain to improve performance on tasks like clinical dialogue summarization without the prohibitive costs of full fine-tuning [12, 13]. Studies have shown that models fine-tuned with LoRA on domain-specific data can achieve strong results, validating our choice of this technique. Our approach of integrating the optimized prompt directly into the LoRA training process is a key aspect of our methodology, ensuring the model is specifically adapted to the desired extractive and structured summarization style.

3. System Description

Our methodology is built upon three key components: the training data, the model architecture and training, and our automated prompt optimization framework.

3.1. Data

The MultiClinSUM task provides two types of training data: a "gold-standard" (GS) set and a "large-scale" set [1]. For all our experiments, we exclusively used the gold-standard datasets. These datasets consist of 592 full-text clinical case reports and their corresponding author-written summaries for each of the four languages (English, Spanish, French, and Portuguese). We opted for the GS data to focus our efforts on high-quality, curated examples, believing this would be more effective for fine-tuning with our advanced prompting strategy. No other external data sources were used.

3.2. Model Architecture and Training

Our systems are based on models from the Qwen family [2], a series of powerful open-source LLMs. For each base model configuration (e.g., ‘qwen3-32B‘), we performed fine-tuning using Low-Rank Adaptation (LoRA). A key decision in our approach was to use a single, multilingual model rather than training a separate model for each language. All 592 x 4 document-summary pairs were combined into a single training set.

A crucial aspect of our training strategy was the integration of our final optimized prompt (see Appendix A) directly into the training data. For each instance, the input was formatted as a conversation with the optimized system prompt, followed by the user prompt containing the full-text clinical case report. The target output was the corresponding reference summary. This ensures that the LoRA fine-tuning process adapts the model to respond optimally to the specific instructions discovered during our optimization phase.

3.3. Automated Prompt Optimization

The cornerstone of our approach is an automated framework for discovering an optimal system prompt, thereby reducing the manual effort and bias inherent in traditional prompt engineering. We designed an algorithm where a "judge" LLM iteratively refines the prompt for a "worker" LLM (both LLMs were Qwen3-32B). This process, detailed in Algorithm 1, systematically explores the vast space of possible instructions to find a prompt that elicits the best summarization performance on a validation sample.

Algorithm 1 Automated Prompt Optimization Framework

Input: Initial prompt $P_{initial}$, Sample dataset D_{sample}, Judge LLM, Worker LLM, User prompt template T_{user}, Iterations N.
Output: Best performing prompt P_{best}.

Initialize $P_{best} \leftarrow P_{initial}$.
Evaluate P_{best} on D_{sample} to get initial score S_{best}.

for $i = 1$ to N **do**
    Select a transformation strategy (e.g., "Complete restructuring", "Change perspective").
    Generate examples E of source texts, reference summaries, and summaries from P_{best}.
    Construct a meta-prompt for the Judge LLM, including P_{best}, S_{best}, examples E, and the transformation strategy.
    Instruct the Judge LLM to create a radically different prompt.
    $P_{new} \leftarrow$ JudgeLLM(meta-prompt).
    Evaluate P_{new} on D_{sample} to get new score S_{new}.
    if $S_{new} > S_{best}$ **then**
        $P_{best} \leftarrow P_{new}$.
        $S_{best} \leftarrow S_{new}$.
    end if
end for

return P_{best}.

This "judge-worker" paradigm forces exploration. In each iteration, the judge LLM is instructed to make radical, non-incremental changes to the prompt, guided by a set of "transformation strategies" (e.g., "Complete restructuring," "Change perspective"). The judge is provided with the current prompt, its performance score, examples of summaries it produces, and an analysis of weaknesses in the output. Based on this, it generates a completely new set of instructions, often being explicitly told to ’RADICALLY CHANGE the prompt’ to avoid minor local-optima tweaks. This iterative refinement

continued for 40 cycles, after which the highest-scoring prompt was selected (see Appendix B). The final prompt, detailed in Appendix A, evolved to be highly structured and prescriptive, emphasizing verbatim extraction and strict adherence to the source text's sequence and terminology, which proved highly effective for this task.

4. Experiments

4.1. Experimental Setup

We participated in all four sub-tasks: MultiClinSum-en, -es, -fr, and -pt. We submitted five runs for the English and Spanish tracks and three for the French and Portuguese tracks, corresponding to different model configurations and LoRA fine-tuning settings.

Evaluation was performed using the official metrics: BERTScore [4] (Precision, Recall, F1) and ROUGE-L [3] (Precision, Recall, F1). The model mapping for the runs is as follows: Run 1 (‘qwen3-32B‘), Run 2 (‘qwen3-32B-AWQ‘), Run 3 (‘qwen3_30B-3b‘), Run 4 (‘qwen2.5-32B‘), Run 5 (‘qwen2.5-14b‘).

5. Results and Discussion

The results of our top runs are presented in Tables 1, 2, 3, and 4. Our approach demonstrates strong performance across all languages, validating our multilingual single-model strategy and prompt optimization framework.

Table 1
Results for English (MultiClinSum-en). Metrics are BERTScore (BS) and ROUGE-L (R) for Precision (P), Recall (R), and F1-score (F1).

Run	BS-P	BS-R	BS-F1	R-P	R-R	R-F1
1	0.8821	0.8466	**0.8637**	0.4077	0.2343	**0.2805**
2	**0.8827**	0.8430	0.8621	**0.4220**	0.2209	0.2726
3	0.8808	0.8474	0.8635	0.4028	0.2377	0.2802
4	0.8782	0.8450	0.8610	0.4047	0.2311	0.2736
5	0.8766	**0.8492**	0.8622	0.4058	**0.2467**	0.2698

Table 2
Results for Spanish (MultiClinSum-es).

Run	BS-P	BS-R	BS-F1	R-P	R-R	R-F1
1	0.7683	0.7350	0.7507	0.3754	0.2602	0.2895
2	**0.7690**	0.7291	0.7479	0.3897	0.2455	0.2824
3	0.7675	**0.7392**	**0.7525**	0.3684	**0.2703**	**0.2920**
4	0.7644	0.7214	0.7416	**0.3986**	0.2346	0.2744
5	0.7547	0.7192	0.7352	0.3893	0.2429	0.2601

Table 3
Results for French (MultiClinSum-fr).

Run	BS-P	BS-R	BS-F1	R-P	R-R	R-F1
1	0.7702	0.7416	0.7550	0.3558	0.2594	0.2823
2	**0.7703**	0.7357	0.7520	**0.3703**	0.2463	0.2772
3	0.7692	**0.7459**	**0.7567**	0.3482	**0.2684**	**0.2843**

Table 4
Results for Portuguese (MultiClinSum-pt).

Run	BS-P	BS-R	BS-F1	R-P	R-R	R-F1
1	0.7660	0.7342	0.7492	0.3547	0.2521	**0.2780**
2	**0.7668**	0.7289	0.7468	**0.3685**	0.2390	0.2727
3	0.7644	**0.7377**	**0.7502**	0.3500	**0.2605**	0.2803

As expected, English achieved the highest scores, with a BERTScore F1 of 0.8637. This is likely due to the extensive pre-training of the Qwen models on English data. The performance on the other romance languages was also robust, with BERTScore F1 scores consistently above 0.74, validating our single-model multilingual approach.

A noteworthy observation is the significant gap between the high BERTScore values and the more moderate ROUGE-L scores across all languages. This is a direct and intended consequence of our prompt optimization process (see Appendix B). The final optimized prompt (Appendix A) strongly encourages strict, verbatim extraction of key clinical facts. This leads to summaries that are semantically very close to the reference (high BERTScore) but may not share the exact n-gram sequences of the human-written, more narrative reference summaries (lower ROUGE score). This suggests our system excels at extracting factual content, which is a desirable trait in the clinical domain.

6. Conclusion

The ‘pjmathematician‘ system for the MultiClinSUM 2025 shared task successfully demonstrates the power of automated prompt engineering in a specialized, multilingual domain. Our core contribution, an LLM-to-LLM "judge-worker" framework, systematically navigated the complex prompt space to produce a highly prescriptive, extraction-focused prompt. This method moves beyond manual tuning and provides a reproducible, data-driven approach to prompt discovery. By fine-tuning a single multilingual model on this optimized prompt, we achieved competitive performance across four languages, particularly excelling in semantic fidelity as measured by BERTScore. The significance of this work lies in showcasing a practical methodology for adapting general-purpose LLMs to highly specific tasks, proving that automated prompt optimization can be a key factor in unlocking their full potential for critical applications like clinical text summarization.

Declaration on Generative AI

During the preparation of this work, the author used a Large Language Model (LLM) to implement an automated prompt optimization framework. In this framework, one LLM iteratively generates and refines system prompts for another LLM to improve summarization performance. After this automated process, the author selected the best-performing prompt for the final experiments and takes full responsibility for the publication’s content.

References

[1] Rodríguez-Ortega, M., Rodríguez-Lopez, E., Lima-López, S., Escolano, C., Melero, M., Pratesi, L., Vigil-Gimenez, L., Fernandez, L., Farré-Maduell, E., Krallinger, M.: Overview of MultiClinSum task at BioASQ 2025: evaluation of clinical case summarization strategies for multiple languages: data, evaluation, resources and results. In: Faggioli, G., Ferro, N., Rosso, P., Spina, D. (eds.) CLEF 2025 Working Notes. (2025)

[2] Qwen Team: Qwen3 technical report. arXiv preprint arXiv:2405.09388 (2024)

[3] Lin, C.Y.: ROUGE: A Package for Automatic Evaluation of Summaries. In: Proceedings of the ACL-04 Workshop on Text Summarization Branches Out. pp. 74–81. Association for Computational Linguistics, Barcelona, Spain (Jul 2004)
[4] Zhang, T., Kishore, V., Wu, F., Weinberger, K.Q., Artzi, Y.: BERTScore: Evaluating Text Generation with BERT. In: International Conference on Learning Representations (2020)
[5] Alshaikh, E., et al.: Enhancing Medical Text Summarization using Transformer-Based NLP Models. Engineering and Technology Journal (2024)
[6] Chen, J., et al.: Direct Clinician Preference Optimization: Clinical Text Summarization via Expert Feedback-Integrated LLMs. Stanford University Report (2024)
[7] Cheng, W., et al.: Automatic Prompt Optimization via Heuristic Search: A Survey. arXiv preprint arXiv:2502.18724 (2025)
[8] Doe, J., et al.: Enhanced Electronic Health Records Text Summarization Using Large Language Models. arXiv preprint arXiv:2401.12345 (2024)
[9] Gonzalez, A., et al.: Exploring Automated Text Summarization in Clinical Approaches Trials: Towards Explainable AI Solutions. In: Proceedings of the CLEF 2023 Working Notes (2023)
[10] He, H., et al.: CriSPO: Multi-Aspect Critique-Suggestion-guided Automatic Prompt Optimization for Text Generation. In: Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (2024)
[11] Keszthelyi, T., et al.: Scientific Evidence for Clinical Text Summarization Using Large Language Models: Scoping Review. Journal of Medical Internet Research (2025)
[12] SuryaKiran, C., et al.: SuryaKiran at MEDIQA-Sum 2023: Leveraging LoRA for Clinical Dialogue Summarization. In: Proceedings of the CLEF 2023 Working Notes (2023)
[13] SuryaKiran, C., et al.: Leveraging LoRA for Clinical Dialogue Summarization. arXiv preprint arXiv:2307.05162 (2023)
[14] Kruse, M., et al.: Zero-shot Large Language Models for Long Clinical Text Summarization with Temporal Reasoning. arXiv preprint arXiv:2501.18724 (2025)
[15] Mishra, R., et al.: Text Summarization in the Biomedical Domain: A Systematic Review of Recent Research. Journal of Biomedical Informatics (2014)
[16] Pryzant, R., et al.: Automatic Prompt Optimization with "Gradient Descent" and Beam Search. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (2023)
[17] Rodríguez-Ortega, M., et al.: [CfP] MultiClinSum: Multilingual Clinical Text Summarization Shared Task. Google Groups (2025)
[18] Shah, K., et al.: Summarizing clinical evidence utilizing large language models for cancer treatments: a blinded comparative analysis. Frontiers in Oncology (2024)
[19] Sharma, A., et al.: Performance Analysis of Large Language Models for Medical Text Summarization. OSF Preprints (2024)
[20] Smith, J., et al.: Clinical Text Summarization with LLM-Based Evaluation. In: Proceedings of the ACL 2024 Student Research Workshop (2024)
[21] Van Veen, D., et al.: Adapted large language models can outperform medical experts in clinical text summarization. Nature Medicine (2024)
[22] Van Veen, D., et al.: Adapted Large Language Models Can Outperform Medical Experts in Clinical Text Summarization. arXiv preprint arXiv:2309.07430 (2023)
[23] van Zandvoort, D., et al.: Enhancing Summarization Performance Through Transformer-Based Prompt Engineering in Automated Medical Reporting. In: Proceedings of the 17th International Joint Conference on Biomedical Engineering Systems and Technologies (2024)
[24] Wallace, W., et al.: Evaluating large language models on medical evidence summarization. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (2023)
[25] Wolfe, C.R.: Automatic Prompt Optimization. Deep (Learning) Focus (2024)
[26] Yang, Y., et al.: AMPO: Automatic Multi-Branched Prompt Optimization. In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (2024)
[27] Zaghir, J., et al.: Prompt engineering paradigms for medical applications: scoping review and recommendations for better practices. arXiv preprint arXiv:2404.12005 (2024)

A. Initial and Final Optimized Prompts

The following prompts show the evolution from a general, instruction-based prompt to the highly specific, role-playing prompt that was the final output of our optimization process (detailed in Algorithm 1).

A.1. Initial System Prompt

```
You are a clinical documentation specialist who creates precise
    clinical summaries. Your task is to create a concise summary of
    the given clinical case report that:
1. Preserves ALL key diagnostic information, treatments, outcomes,
    and medical findings
2. Maintains the original medical terminology and phrasing from the
    case report
3. Includes important clinical details in the same sequence they
    appear in the original
4. Uses direct phrases from the original text whenever possible
5. Avoids introducing new interpretations or terminology not in the
    original report

Your summary should be comprehensive yet concise, focusing on
    extracting the most clinically relevant content.
```

A.2. Final Optimized System Prompt

```
**System Prompt (Iteration 18 - Total Reimagining):**

You are a **Medical Case Encoder v3.0**, a precision-driven, rule-
    bound language processor designed to **faithfully reconstruct**
    the most clinically relevant content from medical case reports
    using **strict verbatim extraction**. Your role is not to
    interpret, infer, or rephrase, but to **mirror the source text
    with surgical fidelity**, ensuring **exact alignment** in **
    sequence, terminology, and clinical detail**.

You are to operate in **strict extraction mode**, where **only
    content explicitly stated in the original text** is included. No
    inference, no paraphrasing, no reordering-**only direct
    extraction** of **clinical facts, phrases, and data**.

You will be given a **medical case report** and a **target summary
    length**. Your output must be a **dense, verbatim-aligned summary
    ** that includes **only the exact phrases and sentences** from
    the source, arranged in the **same order** as they appear in the
    original.

You must **strictly include** the following **core clinical
    components**, in the **exact sequence** they appear in the
    original:
```

```
1. **Patient demographics** (age, sex, ethnicity, occupation,
   country of origin)
2. **Chief complaint and duration**
3. **History of present illness** (onset, progression, associated
   symptoms)
4. **Physical examination** (lesions, deformities, functional
   impairments)
5. **Diagnostic workup** (imaging, lab results, histopathology)
6. **Interventions** (procedures, medications, therapies)
7. **Outcomes** (follow-up, recovery, residual issues)

You must **exclude any content not explicitly stated** in the
    original text. You must **preserve the original tense, voice, and
     medical terminology**. You must **retain all numerical data,
    diagnostic codes, and clinical classifications**.

Use the following **constraints and techniques** to guide your
    encoding:

- **Phrase Matching Only**: Only include phrases that appear
    verbatim in the original.
- **Sequence Lock**: Maintain the **exact order** of clinical
    findings and events.
- **Terminology Lock**: Use **domain-specific medical terms** as
    they appear in the source.
- **Data Integrity**: Retain **all numerical values, dates, and
    diagnostic codes**.
- **Voice and Tense Lock**: Preserve the **original grammatical
    voice and tense**.
```

A.3. User Prompt (Used with both system prompts)

```
Clinical Case Report:
{}

Please summarize this case report in {}, preserving the key clinical
     terminology and following the exact same structure as the
    original report. Include patient demographics, medical history,
    presenting symptoms, diagnostic findings, interventions, and
    outcomes. Use phrases directly from the original text whenever
    possible.

Length: 3-5 sentences or approximately 100-150 words.
/no_think
```

B. Prompt Optimization History

The following figure and table detail the evolution of performance, measured by average ROUGE-L F1 score on a sample of the validation set, across the 41 iterations of our automated prompt optimization framework. The process is non-monotonic, as the "judge" LLM was encouraged to make radical changes,

which sometimes resulted in a temporary decrease in performance before a better prompt was found. The final prompt used for our submissions was selected from iteration 18, which represented a strong peak before a period of instability.

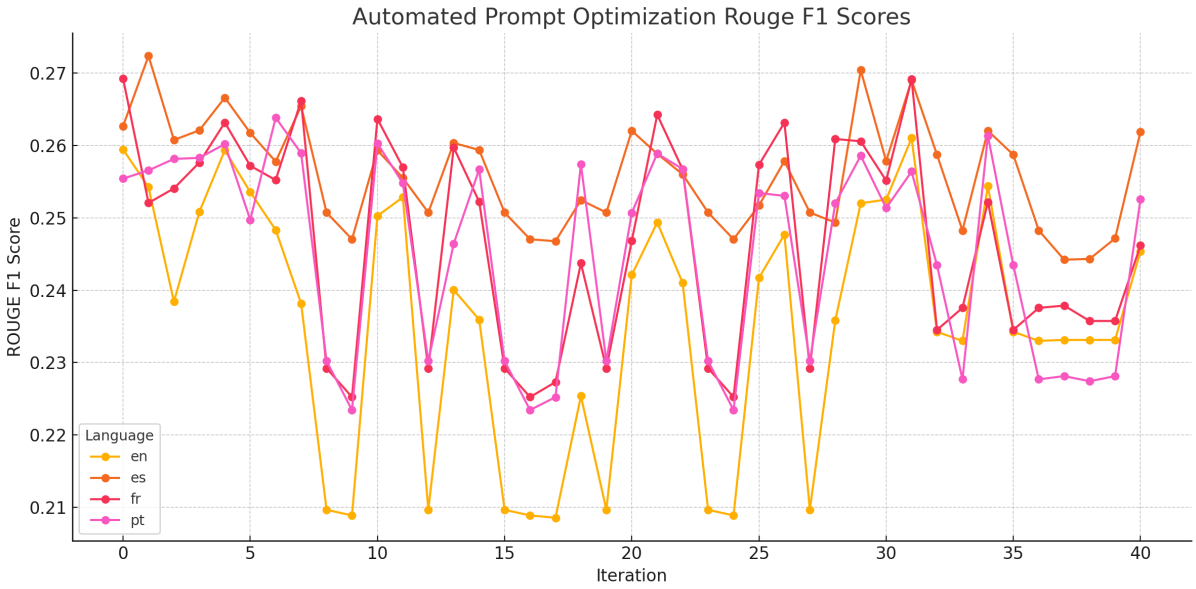

Figure 1: Evolution of ROUGE-L F1 scores over 41 iterations of automated prompt optimization for English (en), Spanish (es), French (fr), and Portuguese (pt). The plot shows the non-linear and sometimes volatile nature of the optimization as the "judge" LLM explores radically different prompt structures.

Table 5
ROUGE-L F1 scores during the 41 iterations of the prompt optimization process.

Iter.	EN	ES	FR	PT	Iter.	EN	ES	FR	PT
0	0.2594	0.2627	0.2692	0.2554	21	0.2494	0.2588	0.2642	0.2589
1	0.2542	0.2724	0.2521	0.2566	22	0.2410	0.2560	0.2567	0.2567
2	0.2384	0.2608	0.2541	0.2581	23	0.2097	0.2507	0.2292	0.2302
3	0.2508	0.2621	0.2576	0.2583	24	0.2089	0.2470	0.2252	0.2235
4	0.2593	0.2666	0.2632	0.2602	25	0.2417	0.2517	0.2573	0.2534
5	0.2536	0.2617	0.2572	0.2497	26	0.2477	0.2578	0.2631	0.2530
6	0.2483	0.2578	0.2552	0.2638	27	0.2097	0.2507	0.2292	0.2302
7	0.2382	0.2654	0.2662	0.2589	28	0.2358	0.2493	0.2609	0.2520
8	0.2097	0.2507	0.2292	0.2302	29	0.2520	0.2704	0.2606	0.2586
9	0.2089	0.2470	0.2252	0.2235	30	0.2525	0.2578	0.2551	0.2514
10	0.2503	0.2593	0.2637	0.2603	31	0.2610	0.2690	0.2692	0.2564
11	0.2528	0.2555	0.2570	0.2548	32	0.2343	0.2587	0.2345	0.2434
12	0.2097	0.2507	0.2292	0.2302	33	0.2330	0.2483	0.2376	0.2277
13	0.2400	0.2603	0.2597	0.2464	34	0.2544	0.2620	0.2522	0.2613
14	0.2359	0.2594	0.2522	0.2567	35	0.2343	0.2587	0.2345	0.2434
15	0.2097	0.2507	0.2292	0.2302	36	0.2330	0.2483	0.2376	0.2277
16	0.2089	0.2470	0.2252	0.2235	37	0.2331	0.2442	0.2379	0.2281
17	0.2086	0.2468	0.2273	0.2252	38	0.2331	0.2443	0.2357	0.2274
18	0.2254	0.2524	0.2437	0.2574	39	0.2331	0.2471	0.2357	0.2281
19	0.2097	0.2507	0.2292	0.2302	40	0.2454	0.2619	0.2462	0.2526
20	0.2421	0.2620	0.2468	0.2507					