Chenyang Zhao

zhaochenyang20@gmail.com ⋄ Blog

EDUCATION EXPERIENCE

Department of Computer Science and Technology, Tsinghua University 2020.09 - Present

Bachelor's degree in progress

· **Academic performance: GPA 3.95/4.00, rank 8/182**

· **Language Proficiency: CET-4 649/710**

RESEARCH EXPERIENCE

Large Scale Language Model Pre-training Group at SenseTime 2022.04 - Present

Internship in Natural Language Processing

· Focus on Few-Shot In-Context Learning, Model Security and Chain of Thought Instruction Finetune

DISCOVER Research Group, Institute of Intelligent Industry, Tsinghua University 2021.09 - 2022.04

Internship in Computer Vision

· Top 10 of over 200 units worldwide in the 2nd Jittor Artificial Intelligence Challenge

PROJECT EXPERIENCE

Sabor Requirements Tracking Management System 2022.03 - 2022.07

Project Leader

· Related project at finished product show and concept promo

TsingAnswer Platform 2021.01 - present

Project Leader

· Attract more than 3,000 users, and the average daily visits during the semester is more than 20,000

SOCIAL WORK

Tsinghua University Computer Science and Technology Association 2021.06 - present

General Chairman

· Leader of Summer 2022 Training & Python Data Analysis Course Instructor & DOCS-9 Writer

Tsinghua University Academic Development Center QA Workshop 2022.02 - present

Technical Team Chairman

Tsinghua University Curriculum Advisory Committee 2021.09 - present

Executive member

HONORS

SenseTime Scholarship Most Promising Award Fall 2022

Tsinghua University Comprehensive Merit Scholarship Fall 2022

Tsinghua University Comprehensive Merit Scholarship Fall 2021

TSINGHUA UNIVERSITY
ACADEMIC TRANSCRIPT

Student Name Zhao Chenyang

Gender Male **Student No.** 2020012363 **Student Type** Undergraduate **Date of Admission** September,2020

School/Department Department of Computer Science and Technology **Major** Computer Science and Technology

Course Number	Course Title	Credit	Grade	Point	Year-Semester
10421055	Calculus A(1)	5	A-	4.0	2020-Autumn
10421324	Linear Algebra	4	A	4.0	2020-Autumn
10610183	Ideological Moral and Legal Education	3	A-	4.0	2020-Autumn
10680011	Situation and Policy	1	A-	4.0	2020-Autumn
10691342	Writing and Communication	2	A-	4.0	2020-Autumn
10720011	Physical Education(1)	1	A-	4.0	2020-Autumn
12090052	Military Theory	2	A-	4.0	2020-Autumn
14201092	English for Academic Purposes (A): Spoken Communication	2	A-	4.0	2020-Autumn
24100023	Discrete Mathematics(1)	3	A-	4.0	2020-Autumn
30210041	Introduction to Information Science and Technology	1	B+	3.6	2020-Autumn
30240233	Fundamentals of Programming	3	A-	4.0	2020-Autumn
00701582	InitiationLove, Marriage and Psychology	2	A+	4.0	2021-Spring
10421065	Calculus A(2)	5	A-	4.0	2021-Spring
10421392	Advanced Topics in Linear Algebra (English)	2	A	4.0	2021-Spring
10610193	Outline of Modern Chinese History	3	A	4.0	2021-Spring
10680042	Introduction to Mao Zetong Thought and Socialism with Chinese Characteristics (2)	2	P	N/A	2021-Spring
10720021	Physical Education(2)	1	B	3.3	2021-Spring
14201082	English for Academic Purposes (A): Research Paper Writing	2	A-	4.0	2021-Spring
24100013	Discrete Mathematics(2)	3	A-	4.0	2021-Spring
30240532	Foundation of Object-Oriented Programming	2	A+	4.0	2021-Spring
10430484	Physics for Scientists and Engineers B(1)	4	W	N/A	2021-Spring
12090062	Military Skills	2	B+	3.6	2021-Summer
30240522	Programing and Training	2	A-	4.0	2021-Summer
00781882	Music Phenomena in The Multi-Culture	2	A-	4.0	2021-Autumn
10420252	Introduction to Complex Analysis	2	A+	4.0	2021-Autumn
10430494	Physics for Scientists and Engineers B(2)	4	A-	4.0	2021-Autumn
10610204	Principle of Marxist Philosophy	4	A	4.0	2021-Autumn
10720031	Physical Education(3)	1	A	4.0	2021-Autumn
20240103	Assembly Language Programming	3	A-	4.0	2021-Autumn
30240184	Data Structures	4	B	3.3	2021-Autumn
40240432	Formal Languages and Automata	2	W	N/A	2021-Autumn
10420803	Probability and Statistics	3	P	N/A	2022-Spring
10680022	Introduction to Xi Jinping Thought on Socialism with Chinese Characteristics for a New Era	2	A-	4.0	2022-Spring
10680032	Introduction to Mao Zedong Thoughts and Theoretical System of Socialism with Chinese Characteristic	2	A-	4.0	2022-Spring
10720041	Physical Education(4)	1	A	4.0	2022-Spring
14204222	Word Power Made Easy	2	A-	4.0	2022-Spring
30240042	Introduction to Artificial Intelligence	2	A	4.0	2022-Spring
30240163	Software Engineering	3	A	4.0	2022-Spring
30240343	Digital Logic Circuit	3	P	N/A	2022-Spring
30240551	Digital Logic Experimentation	1	A-	4.0	2022-Spring
40240422	Fundamentals of Computer Graphics	2	A	4.0	2022-Spring
40240912	Theory and Practice of Human Computer Interaction	2	A-	4.0	2022-Spring

Total Credits: 102.0 **GPA:** 3.95 **Date of Graduation:** ********** **Degree Conferred:** *******

Director of Registrar's Office: 尹佳

Official Seal: 清华大学 教务处 学籍管理 业务专用章 (注册中心) (4) 1101081355883

Date Printed: December 9, 2022

 12865E89C955453C4D74

TSINGHUA UNIVERSITY
ACADEMIC TRANSCRIPT

Student Name Zhao Chenyang

Gender Male **Student No.** 2020012363 **Student Type** Undergraduate **Date of Admission** September,2020

School/Department Department of Computer Science and Technology **Major** Computer Science and Technology

Course Number	Course Title	Credit	Grade	Point	Year-Semester
42540023	Students Research Training	3	A	4.0	2022-Spring
40240963	Topics in Quantum Computing	3	A-	4.0	2022-Summer
00701702	Major Issues in the Contemporary World Politics	2	W	N/A	2022-Summer
	**********				

Total Credits: 102.0 **GPA:** 3.95 **Date of Graduation:** ********** **Degree Conferred:** *******

Director of Registrar's Office: 尹佳

Official Seal:

清华大学 业务专用章 (注册中心) (4) 1101081355883

Date Printed: December 9, 2022

12865E89C955453C4D74

KEY TO TRANSCRIPT

I. COURSE NUMBERING SYSTEM

Each course number consists of 8-10 characters.

The first character indicates the course level:

0-4 or H-T, W = undergraduate courses

6-9, A-G or X-Z = graduate courses

II. CREDIT

Credit is reported in terms of semester hours, whether earned during a 16-week semester or a summer session. For 1 unit of credit, either one hour per week is allotted to lecture or discussion, or two hours per week are allotted to laboratory, while more hours are needed for preparation or subsequent reading and study.

III. THE RECORD ENDS WITH **********.

IV. DATE OF GRADUATION and DEGREE CONFERRED

For currently enrolled undergraduates, the columns of DATE OF GRADUATION and DEGREE CONFERRED are **********.

V. GRADING SYSTEMS

a) EFFECTIVE for students who matriculated in spring 2015 and after

(i) Tsinghua University converted to a LETTER GRADING SYSTEM. The table below shows the grades in detail.

(ii) Credits are given for A+, A, A-, B+, B, B-, C+, C, C-, D+, D, P and EX.

(iii) W: Withdrew.

(iv) I: Incomplete. Marked when a student's application is approved for not attending the final exam.

(v) EX: Exemption. Students receive credits for exempted courses.

Grade	Grade Points	Corresponding 100-point Range	Equivalent 100-point value*
A+	4.0	95-100	100
A			98
A-		90-94	92
B+	3.6	85-89	87
B	3.3	80-84	82
B-	3.0	77-79	78
C+	2.6	73-76	75
C	2.3	70-72	71
C-	2.0	67-69	68
D+	1.6	63-66	65
D	1.3	60-62	61
F	0	0-59	0
P	N/A	N/A	N/A
F	N/A	N/A	N/A

* For the transition period in 2015-2018 between the 100-point grading system and the letter grading system, Tsinghua has provided a corresponding average of values in the 100-point range of each grade. The equivalent 100-point value for course receiving credits corresponds to the median in the range. Students who matriculated in spring 2019 and after no longer use the equivalent 100-point value.

b) EFFECTIVE for students who matriculated prior to spring 2015

(i) 100-POINT GRADING SYSTEM: Credits are given for 60 points and above.

(ii) PASS/FAIL SYSTEM: Credits are given for PASS.
DISTINCTION (for undergraduates only): Credits are given for DISTINCTION.

(iii) REPEATED COURSES: The transcript displays only the latest result of a repeated course. Repeated courses are designated with an "Rn" code beside the final grade, where "n" indicates the number of times the course was repeated.

VI. GRADING POLICY REFORM 2015-2018

In the ten years prior to spring 2015, 30 percent of A-range grades have been given. From fall 2015, Tsinghua initiated a grading reform: A-range grades (A+, A, A-) were to account for 20 percent of the grades given in all courses. In Spring 2019, the faculty reaffirmed its commitment to fair and transparent assessment and removed its numeric target for the percent of A-range grades.

VII. GPA CALCULATION

$$\mathrm{GPA} = \frac{\sum \text{Course Credit} * \text{Grade Point}}{\sum \text{Course Credit}}$$

GPA is shown for students who matriculated in spring 2015 and after in a 4.0 grading scale. Course grades with N/A (Not Applicable) should not be included in GPA calculation.

清华大学 计算机科学与技术系
Department of Computer Science and Technology, Tsinghua University

Room 4-504, FIT Building
Tsinghua University,
Beijing, P. R. China, 100084

December 14, 2022

Recommendation Letter for Chenyang Zhao

To Whom It May Concern:

I am Minlie Huang, an Associate Professor in the Computer Science and Technology Department at Tsinghua University. I am writing to highly recommend Chenyang Zhao for a summer research opportunity in your research group. Chenyang is an exceptional student who demonstrated his potential through hard work and dedication.

Chenyang has excellent academic performance. He took my **Object-Oriented Programming Fundamentals** in his first year and performed extraordinarily well with full marks, despite the challenging workload for a freshman student. Moreover, during our after-class discussions, I found that he wrote detailed course notes for all 14 sections and open-sourced them to the Computer Skills Guide Document of Tsinghua University, where they received high appreciation. In his junior year, Chenyang took my **Artificial Neural Networks**. Again, he demonstrated a keen interest in deep learning and excelled at my course assignments.

In addition to his impressive academic performance, Chenyang has demonstrated a strong research interest and exploring spirit, particularly in in-context learning and the security of large-scale language models. He has been concentrating on the Teacher LM, a generalized reasoning model, for his final project in my Artificial Neural Networks class. I frequently notice him engaging graduate students in in-depth discussions regarding the difficulties he overcame in my lab. This project further explores the reasoning ability and interpretability of large-scale language models, and I eagerly await his achievements.

In conclusion, I highly recommend Chenyang for your summer research opportunity. His impressive academic performance, strong interest in research, and down-to-earth dedication to his work make him an outstanding candidate. If you have any further questions concerning Chenyang's application, please do not hesitate to contact me.

Sincerely yours,

Minlie Huang, Associate Professor
Department of Computer Science and Technology
Email: aihuang@tsinghua.edu.cn
Phone: (86) 10 - 6277 - 7699
Web: `http://coai.cs.tsinghua.edu.cn/hml`

商汤 sensetime

No.58 Northwest 4th Ring Road
SenseTime Research,
Beijing, P. R. China, 100080

December 14, 2022

Recommendation Letter for Chenyang Zhao

To Whom It May Concern:

I am Mingjie Zhan, the research leader of the pre-trained language model group at SenseTime Research. I am pleased to recommend our research intern, Chenyang Zhao, for the summer research opportunity in your research team. Chenyang has been working on In-Context Learning and Instruction Finetune since the summer of 2022 and has demonstrated excellent research skills and great potential in this field.

Chenyang has a strong background in deep learning and a talent for quickly learning new concepts and technologies. In the spring of 2022, he led his team to finish in the top 10 of over 200 units worldwide in the 2nd Jittor Artificial Intelligence Challenge. In this competition, he learned the Jittor framework within two weeks, achieving outstanding results. After joining our group, he quickly became proficient in the In-Context Learning and Chain of Thought field. He read over more than 40 essential papers, taking detailed notes and raising valuable questions along the way, demonstrating his critical thinking abilities and research-oriented mindset.

His strong engineering abilities and rigorous work ethic made him well-suited for NLP research, and he made great contributions to our research on the first Generalized Reasoning Model, the Teacher LM. Our work trained the first large-scale language model dedicated to generative reasoning. We utilise the reasoning ability of our model to generate reasoning to existing datasets, resulting in data augmentation. After finetuning on these augmented datasets, other language models perform better on unseen tasks. Chenyang primarily performed the downstream model validation experiment in this process, comparing the effects of finetuning various models on datasets augmented by the Teacher LM. His experiment results showed that the quality of reasoning generated by the Teacher LM exceeds that of the original manual annotations. Our work is currently being submitted to ICML 2023, and Chenyang is a co-first author.

Chenyang's strong learning ability, critical thinking skills, and understanding of research trends make him a valuable addition to any research team. In addition, his meticulous and rigorous work ethic is unmatched. He would make a valuable contribution to your research team, and I highly recommend him for your summer research opportunity. If you have any further questions about him, please do not hesitate to contact me.

Sincerely yours,

Mingjie Zhan

Mingjie Zhan, Research Scientist
SenseTime Research
Email: zhanmingjie@sensetime.com

Room 3-526, FIT Building
Tsinghua University,
Beijing, P. R. China, 100084

December 10, 2022

Recommendation Letter for Chenyang Zhao

To Whom It May Concern:

This is Chun Yu, an Associate Professor in the Department of Computer Science and Technology at Tsinghua University. With pleasure, I am writing this letter of recommendation for Chenyang Zhao, who attended my senior undergraduate course, **Theory and Practice of Human-Computer Interaction**, in the Spring semester of 2022.

Chenyang impressed me with his excellent leadership, collaboration, and professional abilities. In my class, he led a team of 4 students to design a project on intelligent adaptive headphones. They used the underlying communication interface of the Android system to transmit real-time signals and then input the collected user samples into a small-scale neural network to learn users' habits and realize rich user adaptive functions. Chenyang guided his team to achieve adaptive volume adjustment for headphones in two weeks using triple-ended communication.

Chenyang has demonstrated his organizational abilities in various on-campus events. As the project designer, he led the construction of the platform for Tsinghua University's Student Academic Development Association, which is currently used by over 3,000 people daily. In his junior year, as the vice president of the Student Association for Science and Technology at Tsinghua University, Chenyang organized a summer training program at our department, leading 12 peers and individually lecturing the Python and Data Analyses courses. This program popularized Python, Linux, Pytorch, Data Analyses, and other computer skills with praise in over ten sessions, attracting more than 1,000 participants.

Chenyang has had the most outstanding organizational abilities since I joined Tsinghua University as an Associate Professor. I am confident that he will excel in your research team and carry out significant work in the future. For any further information regarding Chenyang's application, please do not hesitate to contact me.

Sincerely yours,

Chun Yu

Chun Yu, Associate Professor
Department of Computer Science and Technology
Email: chunyu@tsinghua.edu.cn
Phone: (86) 10-6277-2471
Web: `http://pi.cs.tsinghua.edu.cn/lab/people/ChunYu/`

Research Proposal

1 Abstract

Despite their notable achievements, there is growing concern about the negative aspects of large pre-trained models. Existing work on large models has analyzed the negative aspects of models through extensive experiments. However, the detailed experimental findings only passively demonstrate the negative aspects of a pre-trained model, and it is difficult to reduce them. The academy has yet to propose a new research paradigm to initiatively control the negative aspects of large-scale language models.

To further utilize the reasoning ability of large models to minimize their negative aspects, this proposal proposes to augment several existing datasets with the forthcoming Generalized Reasoning Model, the Teacher LM, from SenseTime Research. Then augmented datasets will finetune two sets of models with different sizes simultaneously to get the FLAN Model and Verifier. Finally, the negative aspects of the FLAN model in the few-shot inference process will be initiatively controlled by the Verifier combined with the Self-Consistency method.

2 Intuition

Each work on large-scale language models thoroughly analyzed the negative aspects of the models themselves. Earlier models, such as GPT-3 (Brown et al., 2020), tend to have more negative aspects, while newer models have fewer. For example, the negative aspects of Galactica (Taylor et al., 2022) have declined significantly relative to OPT (Zhang et al., 2022) and BLOOM (Scao et al., 2022). However, Galactica's better performance is based on its high-quality scientific task training data. The cost of replacing the training set data from scratch to pre-train new large-scale language models to reduce negative aspects is too high to be applicable. In addition, these studies always appear very passive. The current research paradigm across the academy is that each team analyses the negative aspects of their model in detail. However, the detailed experimental findings only passively demonstrate the negative aspects of a pre-trained model, and it is difficult to reduce them. The academy has yet to propose a new research paradigm to initiatively control the negative aspects of large-scale language models.

Although there is no initiative approach to systematically reduce the negative aspects of large-scale language models, in **Automatically Neutralizing Subjective Bias in Text** (Pryzant et al., 2019), researchers proposed two encoder-decoder baselines to correct for subjective expressions in descriptive language. They also published the **WNC** dataset for research on reducing subjectivity in language. However, their approach only post-processes the generated output, which is also passive and still does not actively mitigate the negative aspects of the model at the time of generation. In **DEXPERTS: Decoding-Time Controlled Text Generation with Experts and Anti-Experts** (Liu et al., 2021), the researchers propose to use Expert and Anti-Expert models to de-toxic the sampling probability of the base model in the inference process. This approach does have the effect of actively reducing the negative aspects of the model during the generation process. However, the authors are not aware of the possible influence of the **Chain of Thought (CoT)** in the model generation process and only stay at direct answer generation.

Further research on **Finetuned Language Models are Zero-Shot Learners(FLAN)** (Wei et al., 2021) proposed that instruction tuning could improve the zero-shot performance on unseen tasks. However, they failed to notice that Chain of Thought may have a more significant effect on the model than instruction in finetuning. In **Scaling Instruction-Finetuned Language Models** (Chung et al., 2022), researchers discussed that only 9 CoT datasets with CoT instruction tuning could effectively reduce the negative aspects of models. But they only added 9 CoT datasets, so it is difficult to show the performance of CoT FLAN finetuned models after adding more CoT datasets. Based on this, in Self-Consistency Improves Chain of Thought Reasoning in Language Models (Wang et al., 2022), self-consistency was recently proposed to significantly improve the performance on unseen tasks by multi-voting several reasoning paths. However, this approach is only effective for close-answer questions, and not all close-answer questions have a valid reasoning path.

Furthermore, though self-consistency significantly improved the model's performance, researchers needed to propose a better way to differentiate each reasoning path's quality further. In Training Verifiers to Solve Math Word Problems(GSM8K) (Cobbe et al., 2021), researchers suggested using the trained Verifier to check the quality of the reasoning generated by the larger model. However, their Verifer performance is not as good as DEXPERTS, and the training process is tedious and difficult.

3 Past Work

In addition to the work already published by other researchers, I am involved in the LLM Research Group at SenseTime Research. In the previous paper Scaling Instruction-Finetuned Language Models, the team significantly improved the model's reasoning ability by adding only 9 instruction tasks with CoT to the finetune. Our work, the first **Generalized Reasoning Model**, called the **Teacher**, started with the construction of more than 2000 instruction tasks with CoT, mainly obtained from various educational companies' English and Chinese test questions, including GRE, SAT, and the Chinese college entrance examination. All tasks consist of high-quality test questions covering humanities, social sciences, medicine, mathematics, linguistics, and so on, with manually annotated analysis as reasoning, related knowledge, and common errors for each question.

Teacher LM's excellent reasoning ability enables it to generate answers, reasonings, related knowledge, and several common errors for numerous given questions. With the Teacher LM, we can augment a large number of small-scale datasets. The input, output, reasoning, related knowledge, and common errors are integrated to obtain a new, augmented dataset. The augmented dataset can then finetune other models with few-shot CoT instructions to perform better on unseen tasks. In addition, augmented dataset finetune can also reduce the negative aspects of the model.

Finally, I will soon submit the first phase of the Teacher LM work to ICML 2023 as a co-first author.

4 Proposal

4.1 Train Verifer and FLAN Model

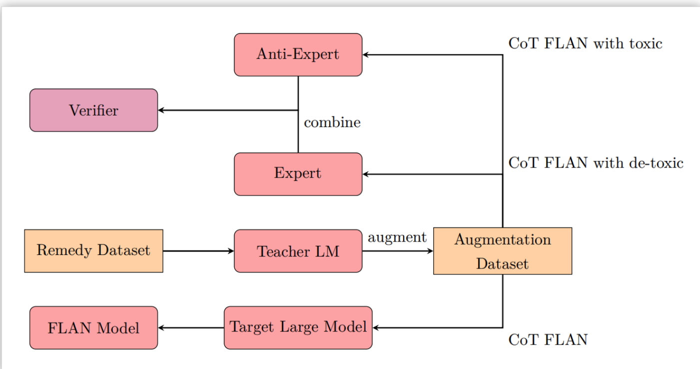

Figure 1: Train Verifer and FLAN Model

In figure 1, we denote the original Model that needs to reduce negative aspects as the **Target Large Model** and introduce two additional Models with smaller parameters labeled **Expert** and **Anti-Expert** (for convenience, we can selecte Facebook/opt-2.7b models as the Expert and Anti-Expert).

Remedy Dataset represents several datasets dedicated to correcting negative aspects, where each instance contains input, toxic output, and de-toxic output. We can create the Remedy Dataset by combining existing datasets, such as the WNC dataset, which provides many modified examples of Wikipedia. Similar instances can also be collected from the Jigsaw Unintended Bias in Toxicity Classification Kaggle challenge. We only need to add instructions to existing datasets to construct the desired Remedy Dataset.

For clarity, we additionally define some relevant concepts. **Instruction Tuning(FLAN)** means that each field of an instance is concatenated using a hint template. We can use the same template for all instances under each dataset. **CoT** stands for Chain of Thought, i.e., a model's thinking process to obtain a particular result. Finally, **CoT FLAN** means that the instances are concurrently concatenated using the template and mixed with the Chain of Thought.

After these demonstrations, we input the Remedy Dataset into the Teacher LM, augment it with the reasoning, related knowledge, and common error of each instance, and construct the **Augmentation Dataset**. Then, we finetune the Large Model using the CoT FLAN approach to obtain the **FLAN Model**. Later, we use the toxic output of the samples in the Augmentation Dataset to finetune the Anti-Expert, and the de-toxic output of the samples to finetune the Expert. Finally, we combine the Expert with the We then combine Expert with Anti-Expert to obtain **Verifier**.

l prop After such finetune, FLAN Model and Verifier have significantly reduced negative aspects compared to their respective base models. And note that the finetune here gets a difference between the Verifier and FLAN Model, where the FLAN Model is mainly used for inference in combination with **Self-Consistency**. In contrast, Verifier is used to correct potentially toxic reasoning in FLAN Model's inference process.

4.2 Self-Consistency and Verify

In figure 2, after obtaining the FLAN Model and Verifier, we design the scheme shown above to further initiatively reduce the negative aspects of the FLAN Model during the reasoning process. But before specifying the scheme, we need to describe how the Verifier works. In the process of reasoning with the FLAN Model, given a prompt $x_{<t}$, the language model computes the logits for the t-th token denoted $\mathbf{z}_t \in R^{|\mathcal{V}|}$, where $\mathcal{V}$ is the vocabulary. A probability distribution over the vocabulary is obtained by normalizing and exponentiating $\mathbf{z}_t$: $P\left(X_t \mid x_{<t}\right) = softmax\left(\mathbf{z}_t\right)$, and the next token is generated by sampling $x_t \sim P\left(X_t \mid x_{<t}\right)$. Next, we label the logits obtained by FLAN Model, Expert, and Anti-Expert as $\mathbf{z}_t, \mathbf{z}_t^{+}$ and $\mathbf{z}_t^{-}$. Finally, the FLAN Model will sample x_t from $\tilde{P}\left(X_t \mid x_{<t}\right) = softmax\left(\mathbf{z}_t + \alpha\left(\mathbf{z}_t^{+} - \mathbf{z}_t^{-}\right)\right)$. We refer to this process as **Verification (Verify, Verifier)**.

After explaining how the Verifier works, when the overall model is inferred, the first step is to input the inference instance into Teacher LM to get the corresponding reasoning, related knowledge, and common errors. Then, we integrate these into **Augmented Input** with CoT instruction and input into FLAN Model to get unique reasoning according to **Greedy Decode**.

Note that the original Self-Consistency approach solves problems such as math word problems and common sense problems, both of which are easy to generate reasoning. However, tasks such as translation, prone to have negative aspects, are not easy to generate valid reasoning. Therefore, we need to verify the validity of the unique reasoning from the greedy decode.

Suppose the unique reasoning obtained by greedy decode is invalid. In that case, we still use the greedy decode method to decode the output directly using the FLAN Model while adding the Verifier to continuously verify the whole decoding process.

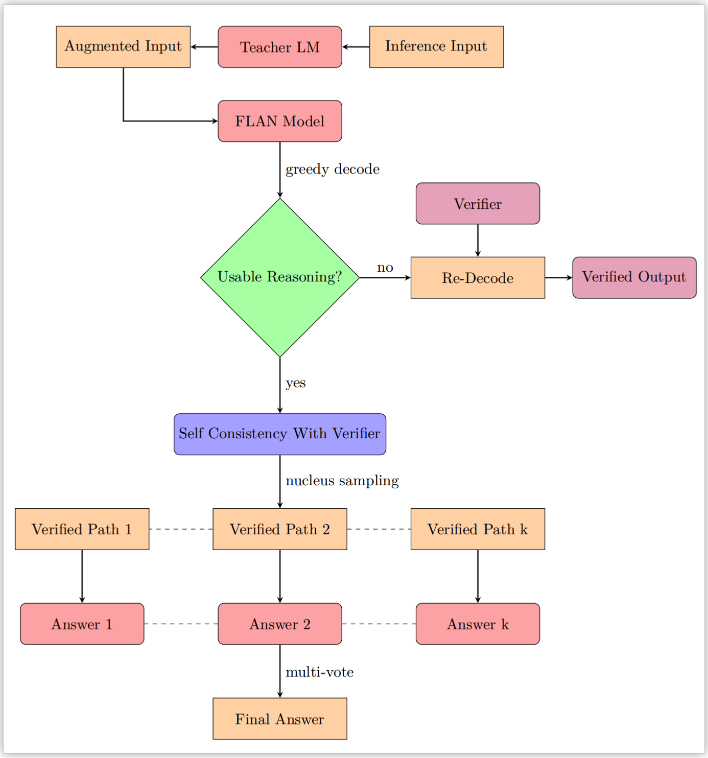

Figure 2: Self-Consistency and Verify

If the unique reasoning obtained by the greedy decode is valid, then the Self-Consistency method is applied. We replace the decode method of the FLAN Model with **Nucleus Sampling** and use Verifier to verify sampling probability constantly. Finally k sampling yields k reasoning as $r_i, i = 1, 2, 3 \cdots k$. $\mathbf{r_i^{ver}}$ with Augmented Augmented Input with CoT is input into FLAN Model again to obtain an answer $\mathrm{a_i^{ver}}$ for each $\mathrm{r_i^{ver}}$ and form an **Answer Set A**. Finally, we multi-vote all k answers in the A set, and the final output is the answer with the most occurrences.

5 Further Discussion

1. Why use the Verifier to correct the Potential Toxic Output and r_i instead of using the FLAN Model itself?

   Training Verifiers to Solve Math Word Problems (Cobbe et al., 2021) mentions that only a tiny Verifier model is needed to achieve an excellent correction compared to the Large Model, and the Expert inside the Verifier is more capable of generating de-toxic sampling possibility than the FLAN Model. In addition, it would be computationally expensive to continue using the FLAN Model to correct itself.

2. Self-Consistency is actually for close-answer questions. How to deal with open-answer questions?

   The open-answer often appears on tasks such as translation, and this problem can be solved by using existing mature models for determining equivalence classes. I did not elaborate in the discussion above but introduced an additional model to construct the $\mathbf{A_{uni}}$ set. For each element $\mathrm{a_i^{ver}}$ in the A set, if there is no equivalent element in A_{uni}, then append $\mathrm{a_i^{ver}}$ into A_{uni}. Otherwise, merge $\mathrm{a_i^{ver}}$ with some elements in A_{uni}. After that, the answer with the most occurrences in A_{uni} will be multi-voted out as the final output.

References

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., et al. (2020). Language models are few-shot learners.

Chung, H. W., Hou, L., Longpre, S., Wei, J., et al. (2022). Scaling instruction-finetuned language models.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., et al. (2021). Training verifiers to solve math word problems.

Liu, A., Sap, M., Lu, X., et al. (2021). Dexperts: Decoding-time controlled text generation with experts and anti-experts.

Pryzant, R., Martinez, R. D., Yang, D., et al. (2019). Automatically neutralizing subjective bias in text. *CoRR*, abs/1911.09709.

Scao, T. L., Fan, A., Akiki, C., Pavlick, E., et al. (2022). Bloom: A 176b-parameter open-access multilingual language model.

Taylor, R., Kardas, M., Cucurull, G., Scialom, T., et al. (2022). Galactica: A large language model for science.

Wang, X., Wei, J., Schuurmans, D., Le, Q., et al. (2022). Self-consistency improves chain of thought reasoning in language models.

Wei, J., Bosma, M., Zhao, V. Y., et al. (2021). Finetuned language models are zero-shot learners.

Zhang, S., Roller, S., Goyal, N., Artetxe, M., et al. (2022). Opt: Open pre-trained transformer language models.