Session #20: Retrieval from Memory

Thursday, November 3
CSCI 601.771: Self-supervised Statistical Models

Johns Hopkins University

As I mentioned in the class, **if everyone is on board** I am happy to move the final project presentations earlier before the final exam.

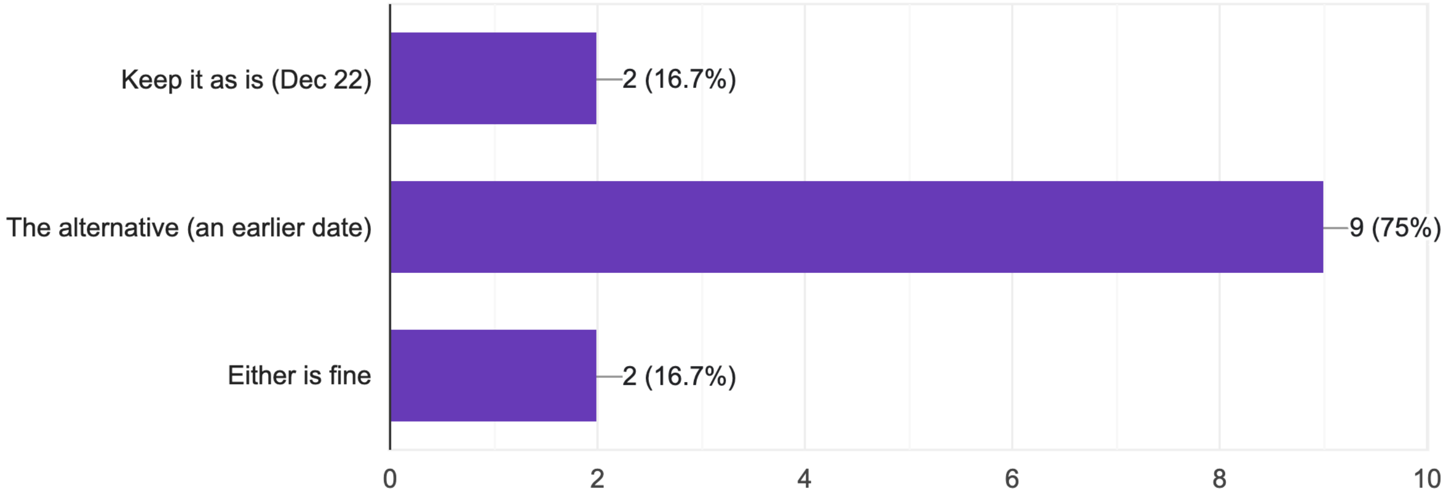
- **Currently**, the final presentation/report is due Dec 22.
- The **alternative** is to move the final presentation to Thursday, Dec 8, and the final report deadline to Sunday, Dec 11 (just before the reading days).

As I mentioned in the class, **if everyone is on board** I am happy to move the final project presentations earlier before the final exam.

- **Currently**, the final presentation/report is due Dec 22.
- The **alternative** is to move the final presentation to Thursday, Dec 8, and the final report deadline to Sunday, Dec 11 (just before the reading days).

What is your preference?

12 responses



Comments

- "I'd prefer the earlier date in general but the only issue is that that week is EMNLP, so a few of us will be traveling. If there is an alternate way to submit our presentations by Dec 8th (e.g. a recording) I'd prefer the earlier date."

- "It would be great if we move the report date to a bit later if possible "

- "[MASK] and I are at EMNLP in Abu Dhabi during this time but we are very happy to do the earlier date + coordinate some sort of virtual presentation if possible "

Problem Statement

Some issues with pre-trained neural language models:

- They cannot easily expand or revise their memory
- They can't straightforwardly provide insight into their predictions
- They may produce "hallucinations"

Stakeholders: Muzzi & Fadil

Model Overview

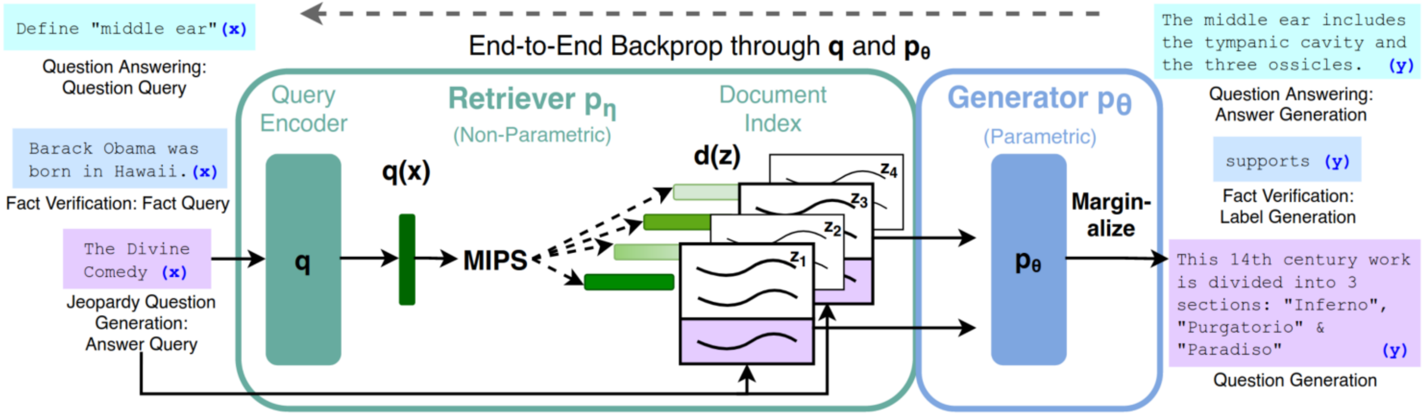


Figure 1: Overview of the model.

- Combined pre-trained retriever (Query Encoder + Document Index) with a pre-trained seq2seq model (Generator) and fine-tune end-to-end.
- For query x, Maximum Inner Product Search (MIPS) is used to find the top-K documents z .

Stakeholders ✍: Muzzi & Fadil

Models difference

$$p_{\text{RAG-Sequence}}(y|x) \approx \sum_{z \in \text{top-}k(p(\cdot|x))} p_\eta(z|x) p_\theta(y|x,z) = \sum_{z \in \text{top-}k(p(\cdot|x))} p_\eta(z|x) \prod_i^N p_\theta(y_i|x,z,y_{1:i-1})$$

$$p_{\text{RAG-Token}}(y|x) \approx \prod_i^N \sum_{z \in \text{top-}k(p(\cdot|x))} p_\eta(z|x) p_\theta(y_i|x,z_i,y_{1:i-1})$$

Both models are trained by directly minimising the log likelihood of each target - log p(y|x)

Stakeholders ✍: Muzzi & Fadil

Overall Setup

- A single Wikipedia dump is used for all experiments
- Each article is split into 100 word chunks to form 21 M documents
- The top k documents are retrieved for each task with k being limited between 5 to 10
- Stakeholders ✍: Muzzi & Fadil

Open Domain Q/A

- Questions are treated as input output text pairs
- RAG is trained by minimizing negative log likelihood
- Comparisons are made to the extractive Q/A paradigm and to closed book Q/A
- Tested on 4 datasets Natural Questions (NQ) , TriviaQA (TQA) WebQuestions (WQ) and CuratedTrec (CT)
- Stakeholders ✍: Muzzi & Fadil

Results

- Sets new SOA performances in all categories
- More efficient than salient span masking training
- Generating answers allows us to get them even when they don't directly exist in the passage

	Model	NQ	TQA	WQ	CT
Closed Book	T5-11B [52]	34.5	- /50.1	37.4	-
	T5-11B+SSM[52]	36.6	- /60.5	44.7	-
Open Book	REALM [20]	40.4	- / -	40.7	46.8
	DPR [26]	41.5	**57.9**/ -	41.1	50.6
	RAG-Token	44.1	55.2/66.1	**45.5**	50.0
	RAG-Seq.	**44.5**	56.8/**68.0**	45.2	**52.2**

Abstractive Q/A

- MMARSCO NLG is used as the task
- There are 10 gold passages retrieved for the questions via a search engine and then annotated into an answer
- Using only the Questions and answers makes MMARSCO abstractive
- RAG must rely on its parametric knowledge to generate reasonable answers
- Stakeholders: Muzzi & Fadil

Results

- Approaches SOA performances

- Specially impressive given that there s no access to the gold passages

- RAG hallucinates less and gives more factually correct answers than BART

Model	Jeopardy B-1	Jeopardy QB-1	MSMARCO R-L	MSMARCO B-1	FVR3 Label Acc.	FVR2 Label Acc.
SotA	-	-	**49.8***	**49.9***	**76.8**	**92.2***
BART	15.1	19.7	38.2	41.6	64.0	81.1
RAG-Tok.	**17.3**	**22.2**	40.1	41.5	72.5	<u>89.5</u>
RAG-Seq.	14.7	21.4	<u>40.8</u>	<u>44.2</u>		

Jeopardy Q/A

- Generates factually demanding Jeopardy questions
- Splits from SearchQA, with 100K train, 14K dev, and 27K test examples are used
- BART model is trained for comparison
- Finally human evaluation is done for accuracy and specificity

- Stakeholders : Muzzi & Fadil

Results

- RAG Token Outperforms RAG Sequence with both outperforming BART

- RAG more factual in 42.7% of cases !

Model	Jeopardy B-1	Jeopardy QB-1	MSMARCO R-L	MSMARCO B-1	FVR3 Label Acc.	FVR2 Label Acc.
SotA	-	-	**49.8***	**49.9***	**76.8**	**92.2***
BART	15.1	19.7	38.2	41.6	64.0	81.1
RAG-Tok.	**17.3**	**22.2**	40.1	41.5	72.5	<u>89.5</u>
RAG-Seq.	14.7	21.4	<u>40.8</u>	<u>44.2</u>		

FEVER

- Classifies whether a natural language claim is supported or refuted by Wikipedia
- Retrieval problem coupled with entailment reasoning task
- Map **FEVER** class labels (supports, refutes, or not enough info) to single output tokens and directly train with **claim-class pairs**
- No supervision is used on retrieved evidence
- Stakeholders: Muzzi & Fadil

Results

- Within 4.3% of SOA systems

- Within 2.7% of RoBERTa SOA although its only given the claim

- Top 10 document are gold in 90% of cases!

Model	Jeopardy B-1	Jeopardy QB-1	MSMARCO R-L	MSMARCO B-1	FVR3 Label Acc.	FVR2 Label Acc.
SotA	-	-	**49.8***	**49.9***	**76.8**	**92.2***
BART	15.1	19.7	38.2	41.6	64.0	81.1
RAG-Tok.	**17.3**	**22.2**	40.1	41.5	72.5	<u>89.5</u>
RAG-Seq.	14.7	21.4	<u>40.8</u>	<u>44.2</u>		

Additional results

1) Generation diversity

	MSMARCO	Jeopardy QGen
Gold	89.6%	90.0%
BART	70.7%	32.4%
RAG-Token	77.8%	46.8%
RAG-Seq.	83.5%	53.8%

2) Retrieval Ablations

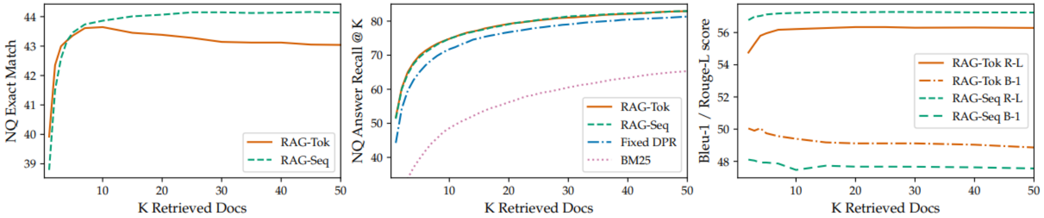
Model	NQ	TQA	WQ	CT	Jeopardy-QGen		MSMarco		FVR-3	FVR-2
	Exact Match				B-1	QB-1	R-L	B-1	Label Accuracy	
RAG-Token-BM25	29.7	41.5	32.1	33.1	17.5	22.3	55.5	48.4	**75.1**	**91.6**
RAG-Sequence-BM25	31.8	44.1	36.6	33.8	11.1	19.5	56.5	46.9		
RAG-Token-Frozen	37.8	50.1	37.1	51.1	16.7	21.7	55.9	49.4	72.9	89.4
RAG-Sequence-Frozen	41.2	52.1	41.8	52.6	11.8	19.6	56.7	47.3		
RAG-Token	43.5	54.8	**46.5**	51.9	**17.9**	**22.6**	56.2	**49.4**	74.5	90.6
RAG-Sequence	**44.0**	**55.8**	44.9	**53.4**	15.3	21.5	**57.2**	47.5		

Additional results

1) Index hot swapping

Replacing non parametric memory is enough to change the way data works in RAG !

1) Effects of more documents



TBD

TBD

TBD

TBD

Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks

Patrick Lewis[†‡], Ethan Perez[⋆],

Aleksandra Piktus[†], Fabio Petroni[†], Vladimir Karpukhin[†], Naman Goyal[†], Heinrich Küttler[†],

Mike Lewis[†], Wen-tau Yih[†], Tim Rocktäschel[†‡], Sebastian Riedel[†‡], Douwe Kiela[†]

[†]Facebook AI Research; [‡]University College London; [⋆]New York University;
`plewis@fb.com`

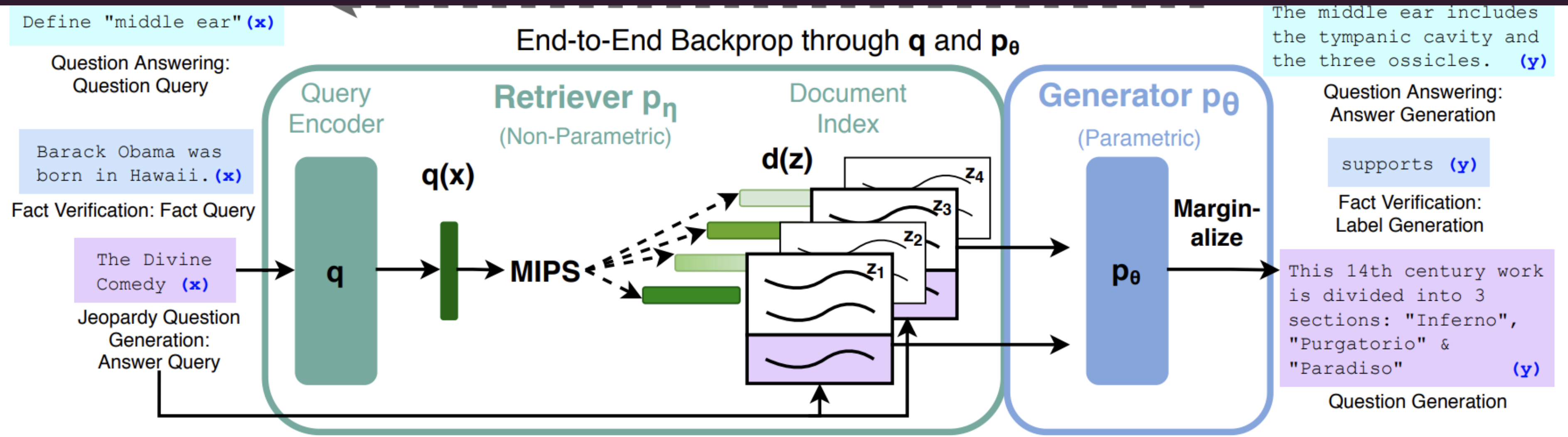
Ayo & Elisée

RAG VERSION DIVERSITY

- Proposed 2 version of RAG:

  RAG Sequence (Retrieved single doc from data base and condition on 1 doc)

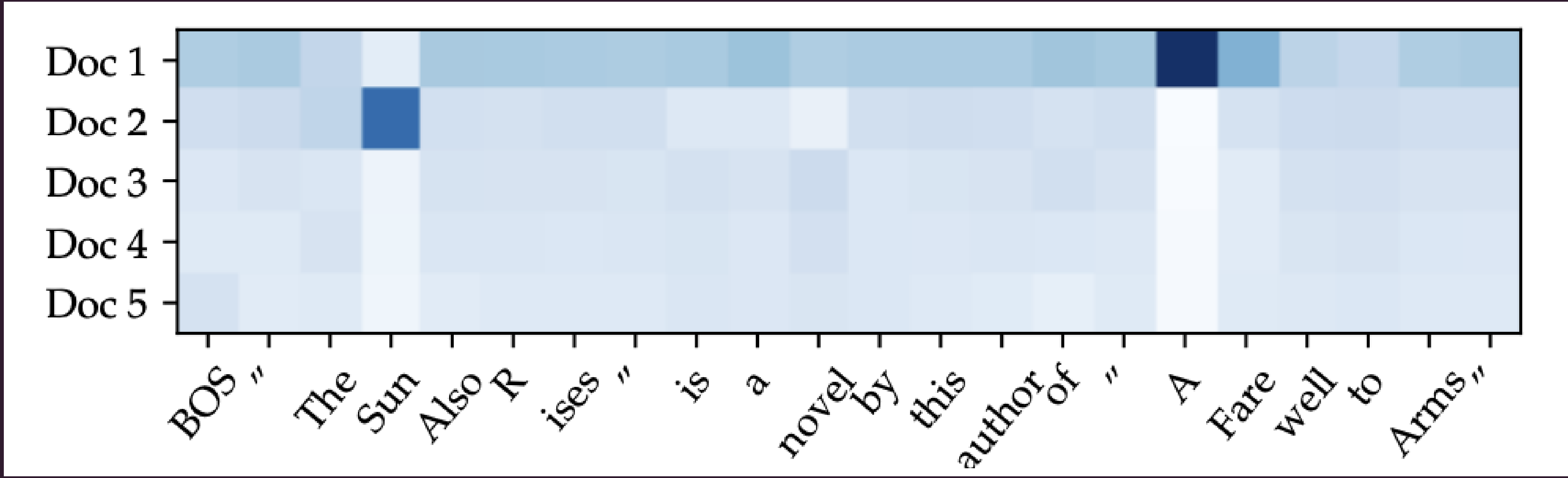
  RAG Token (Retrieved multiple doc from data base and switch b/w the set of doc)



RAG VARIANCE COMPARISON

Model	Jeopardy		MSMARCO		FVR3	FVR2
	B-1	QB-1	R-L	B-1	Label Acc.	
SotA	-	-	**49.8***	**49.9***	**76.8**	**92.2***
BART	15.1	19.7	38.2	41.6	64.0	81.1
RAG-Tok.	**17.3**	**22.2**	40.1	41.5	72.5	<u>89.5</u>
RAG-Seq.	14.7	21.4	<u>40.8</u>	<u>44.2</u>		

▯ INTERACTION B/W PARAMETRIC VS NON-PARAMETRIC



☒ OTHER POSITIVES

- Latent Retrieval: No labels needed for retrieved docs.
- General in use/application: For any seq2seq task.
- Easy to follow and well visualized.

OTHER POSITIVES (CONT.)

- Results indicate what RAG model, RAG-token vs RAG-seq, perform better on certain tasks, so others interested in solving certain problems (Open-domain Question Answering) for instance, might opt to using RAG-seq model.
- Usage of pairwise comparative evaluation as opposed to Likert scores or single-turn pairwise evaluation for Jeopardy Question Generation.
- RAG's world knowledge can be updated by replacing its non-parametric memory unlike GPT (reducing hallucinations).
    - *After updating Wikipedia dumps, they saw that RAG had 70% accuracy in the 2016 index and 68% accuracy in 2018 index. - "Who is the President of Peru?"*

NEGATIVES

- No clear explanation/hypothesis as to why the RAG-Tok model outperforms the RAG-Seq or vice versa in certain experiments. This was only done for the Jeopardy Question Generation experiment but not others.
- Bringing up points not previously addressed in previous sections of the paper, such as the reference to Rouge-L points (could be a negative based on the type of reader).
- Nit-pick: Grammatical errors
- Retrieval and Generation framework discussed in the paper has also been used in other papers but was presented a new idea that was developed.

Retrieval-guided Dialogue Response Generation via a Matching-to-Generation Framework

- Does RAG only look at Wikipedia articles? What happens if these articles are outdated? How can we fact-check the "facts?" This was a downside addressed in the broader impact section of the paper.

RAG Fact-verification experiments

- Goal: Retrieve evidence related to claim, reasoning about this to classify whether supported, refused, or unverified by Wikipedia
- Input: a claim, output: 3 classes
- Test RAG models classification ability
- Map 3 labels to single token and train claim-class pairs

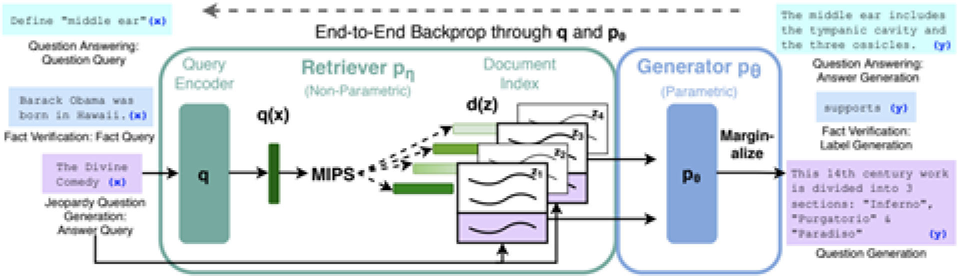


Figure 1: Overview of our approach. We combine a pre-trained retriever (*Query Encoder* + *Document Index*) with a pre-trained seq2seq model (*Generator*) and fine-tune end-to-end. For query x, we use Maximum Inner Product Search (MIPS) to find the top-K documents z_i. For final prediction y, we treat z as a latent variable and marginalize over seq2seq predictions given different documents.

Table 2: Generation and classification Test Scores. MS-MARCO SotA is [4], FEVER-3 is [68] and FEVER-2 is [57] *Uses gold context/evidence. Best model without gold access underlined.

Model	Jeopardy B-1	Jeopardy QB-1	MSMARCO R-L	MSMARCO B-1	FVR3 Label Acc.	FVR2 Label Acc.
SotA	-	-	**49.8***	**49.9***	**76.8**	**92.2***
BART	15.1	19.7	38.2	41.6	64.0	81.1
RAG-Tok.	**17.3**	**22.2**	40.1	41.5	72.5	89.5
RAG-Seq.	14.7	21.4	40.8	44.2		

Patrick, et al., 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks

Use RAG model to support improving Wikipedia Verifiability

- Motivation: claims that are likely to be challenged need to be backed by citations

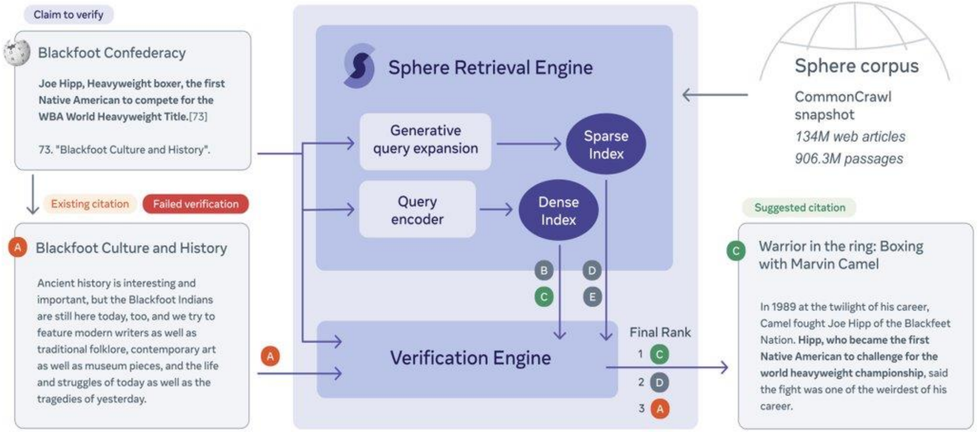


Figure 1: The decision flow of SIDE from a claim on Wikipedia to a suggestion for a new citation is as follows: (1) the claim is sent to the *Sphere Retrieval Engine* which produces a list of potential candidate documents from the *Sphere corpus*; (2) the *verification engine* ranks the candidate documents and the original citation w.r.t. the claim; (3) if the original citation is not ranked above the candidate documents, then a new citation from the retrieved candidates is suggested. Note that the score of the *verification engine* can be indicative of a potential *failed verification*, as the one reported in the example.

Fabio, et al., 2022. Improving Wikipedia Verifiability with AI