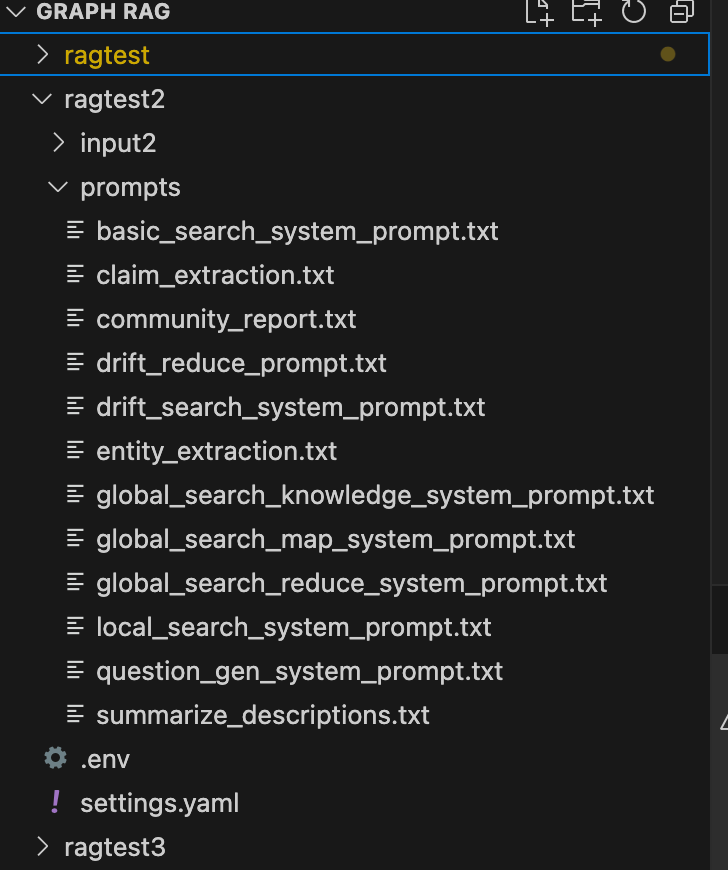
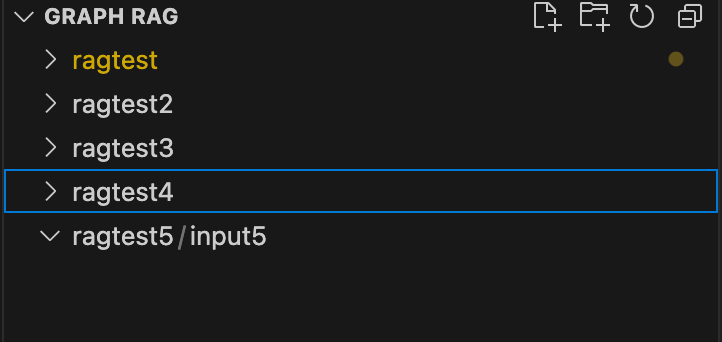
**Procedure of auto tuning prompts**

**Step 1: initialize graphrag and it will create automatic prompts.**

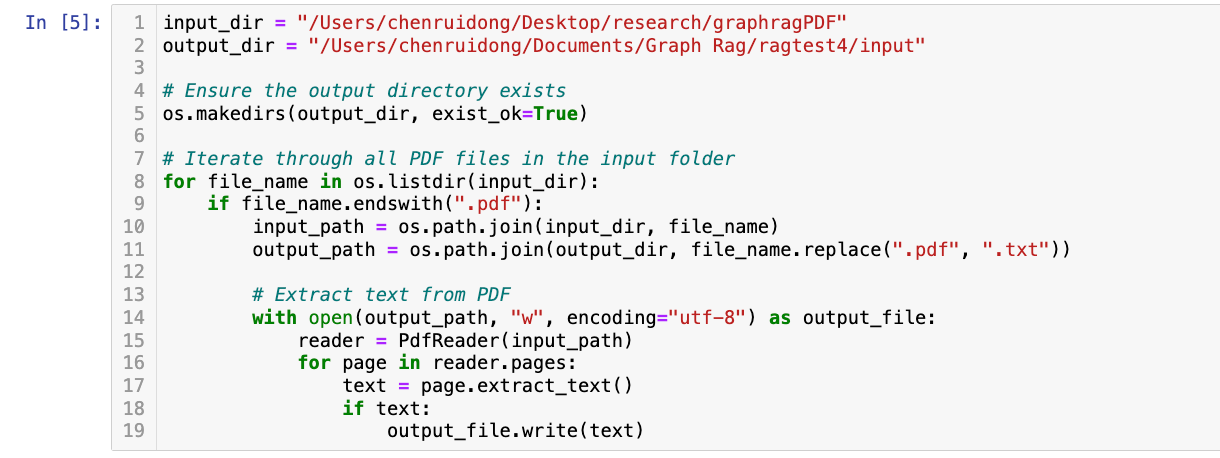
**graphrag init --root ./ragtest**



**Step 2: create some folders other than ragtest so we can tune our own prompts inside these folders independently.**



**Step 3: auto tune the prompts (core step) and it will create 3 new prompts based on domain txt files. We need to give the input files first, or it will get errors. The auto tuning method will use input files. Besides, we need to set the parameters inside .env files and .settings.yaml file. It will combine llm model and input txt files to generate the auto tuning prompts. insert the txt files into the input folder. We can transform the PDF files into txt files based on some function.**



**(1): Default settings, no auto-tuning prompts**

**591 nodes; 562 relationships; 80 communities**

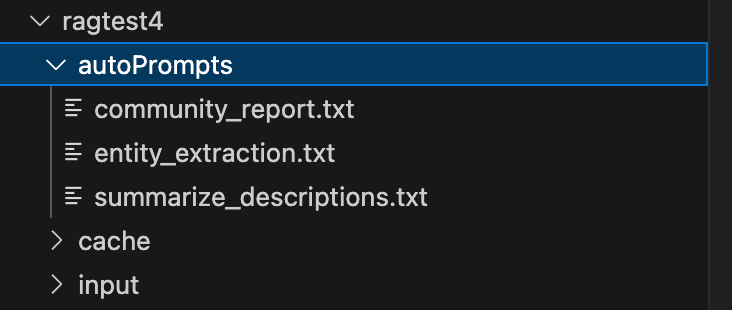
**(2): Ragtest4**

python -m graphrag prompt-tune --root ./ragtest4 --config ./ragtest4/settings.yaml --domain "**biomedicine**" \

--selection-method random --limit **10** --language English --max-tokens **2048** --chunk-size **256** --min-examples-required **3** \

--discover-entity-types --output ./ragtest4/autoPrompts

**1062 nodes; 1677 relationships; 80 communities**



**(3): Ragtest5**

python -m graphrag prompt-tune --root ./ragtest5 --config ./ragtest5/settings.yaml --domain "**biomedicine**" \

--selection-method random --limit **15** --language English --max-tokens **4000** --chunk-size **512** --min-examples-required **4** \

--discover-entity-types --output ./ragtest5/autoPrompts

**223 nodes; 281 relationships; 34 communities**

**(4): Ragtest2**

python -m graphrag prompt-tune --root ./ragtest2 --config ./ragtest2/settings.yaml --domain "**biomedicine**" \

--selection-method random --limit **15** --language English --max-tokens **4000** --chunk-size **256** --min-examples-required **4** \

--discover-entity-types --output ./ragtest2/autoPrompts

**325 nodes; 370 relationships; 57 communities**

**(5): Ragtest3**

python -m graphrag prompt-tune --root ./ragtest3 --config ./ragtest3/settings.yaml --domain "**biomedicine**" \

--selection-method random --limit **15** --language English --max-tokens **2048** --chunk-size **512** --min-examples-required **4** \

--discover-entity-types --output ./ragtest3/autoPrompts

**297 nodes; 357 relationships; 46 communities**

1. **: Ragtest6 - rerun ragtest2, same codes with errors**

python -m graphrag prompt-tune --root ./ragtest9 --config ./ragtest9/settings.yaml --domain "**biomedicine**" \

--selection-method random --limit **15** --language English --max-tokens **4000** --chunk-size **256** --min-examples-required **4** \

--discover-entity-types --output ./ragtest9/autoPrompts

1. **: Ragtest7**

python -m graphrag prompt-tune --root ./ragtest7 --config ./ragtest7/settings.yaml --domain "**biomedicine**" \

--selection-method random --limit **15** --language English --max-tokens **2048** --chunk-size **256** --min-examples-required **3** \

--discover-entity-types --output ./ragtest7/autoPrompts

**625 nodes; 596 relationships; 81 communities**

1. **: Ragtest8**

python -m graphrag prompt-tune --root ./ragtest8 --config ./ragtest8/settings.yaml --domain "**biomedicine**" \

--selection-method random --limit **10** --language English --max-tokens **2048** --chunk-size **256** --min-examples-required **4** \

--discover-entity-types --output ./ragtest8/autoPrompts

**666 nodes; 1205 relationships; 46 communities**

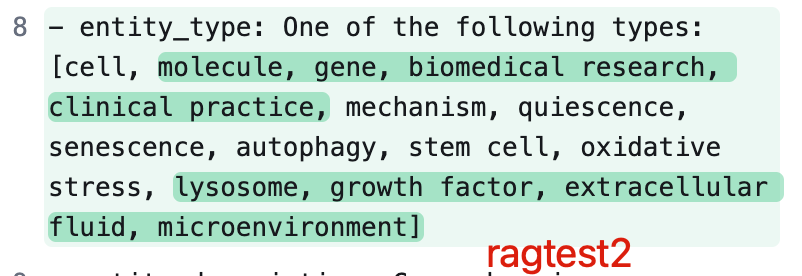
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **domian** | **limit** | **Max-tokens** | **Chunk size** | **Min examples** | **Nodes number** | **Relationships**  **numbers** | **Communities number** | **#** |
| No auto tuning prompts | NA | NA | NA | NA | 591 | 562 | 80 | 1,ragtest |
| Biomedicine | 10 | 2048 | 256 | 3 | 1062 | 1677 | 80 | 2,ragtest4 |
| Biomedicine | 15 | 4000 | 512 | 4 | 223 | 281 | 34 | 3,ragtest5 |
| Biomedicine | 15 | 4000 | 256 | 4 | 325 | 370 | 57 | 4,ragtest2 |
| Biomedicine | 15 | 2048 | 512 | 4 | 297 | 357 | 46 | 5,ragtest3 |
| Biomedicine | 15 | 2048 | 256 | 3 | 625 | 596 | 81 | 6,ragtest7 |
| Biomedicine | 10 | 2048 | 256 | 4 | 367 | 386 | 46 | 7,ragtest8 |

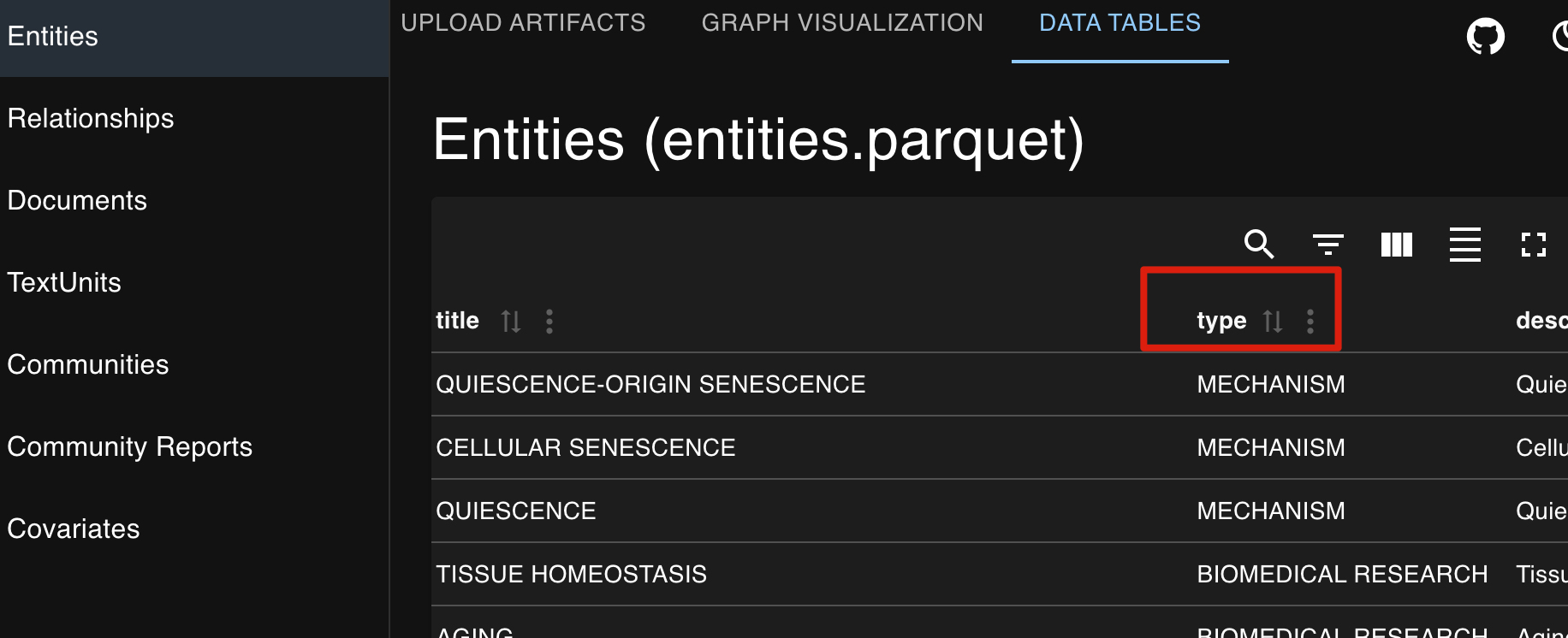
**Compare 2 and 7, increase minimum examples, the number of nodes and relationships and communities numbers decrease.**

**Check the difference about entities prompts via Diffchecker**

**Compare 2 and 6, increase limit, the number of nodes and relationships decrease and the communities number keeps similar**

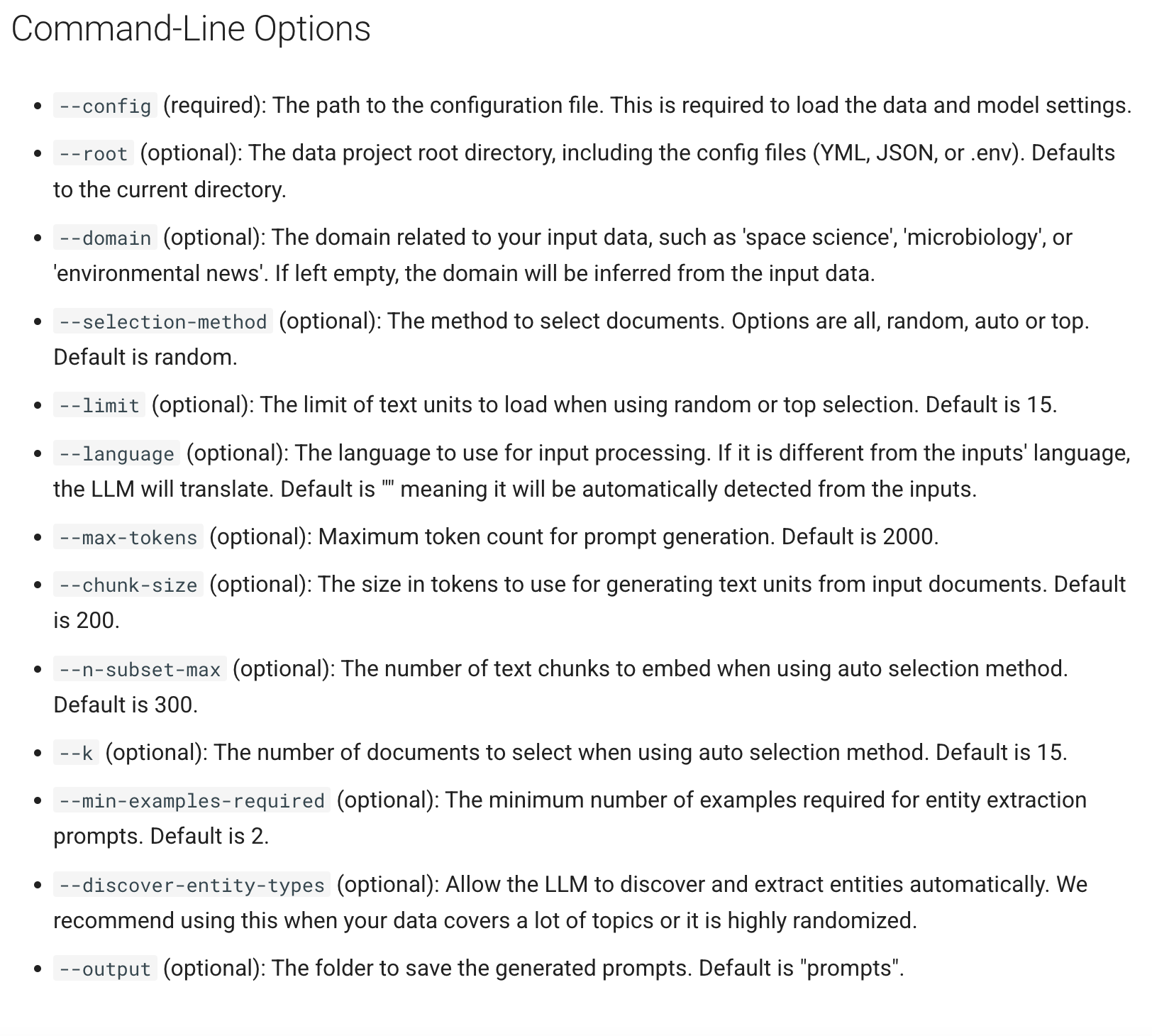
**Compare 3 and 4, increase chunk size, the number of nodes and relationships and communities decrease**

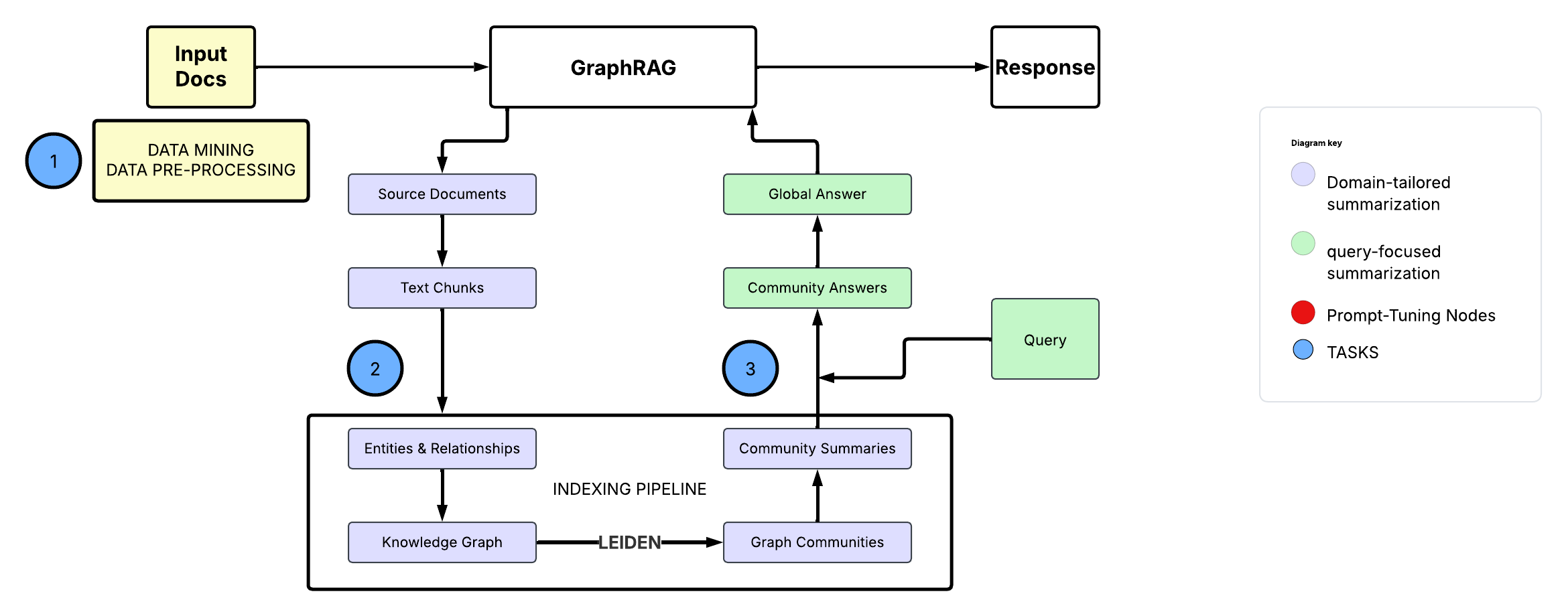




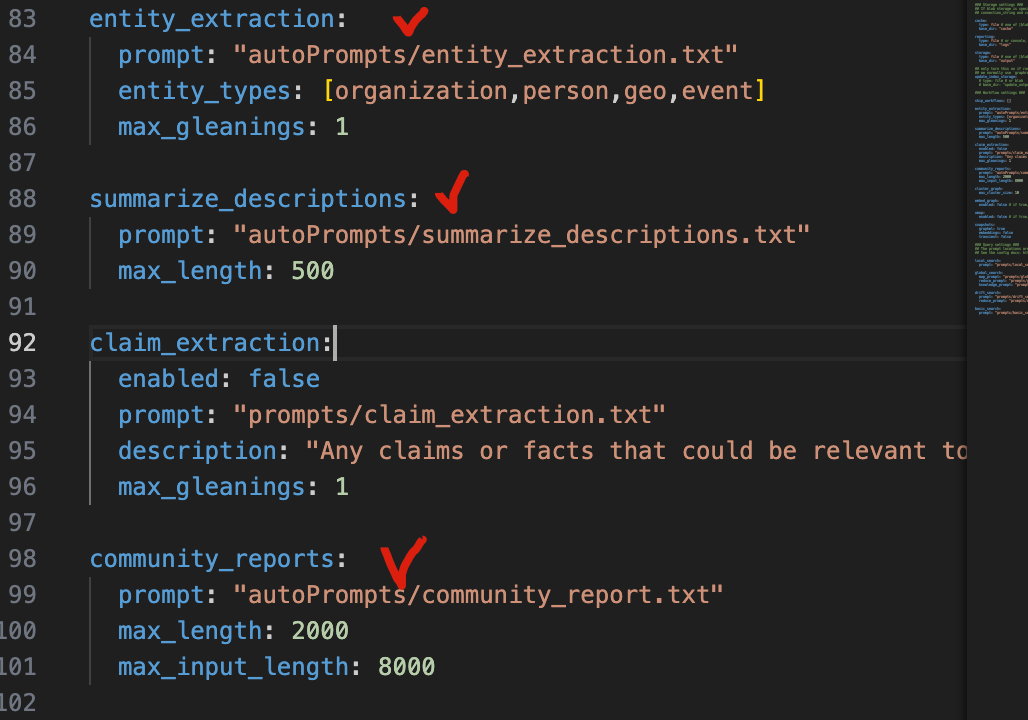
**Compare 3 and 5, increase max-tokens, the number of nodes and relationships and communities decrease.**

**To have more nodes, more relationships, more communities, we should set smaller minimum examples, smaller limit, smaller chunk size, smaller max tokens.**





**Step 4: update the configuration in settings.yaml file so that later when analyze our given input, the model will use these 3 customized prompts.**



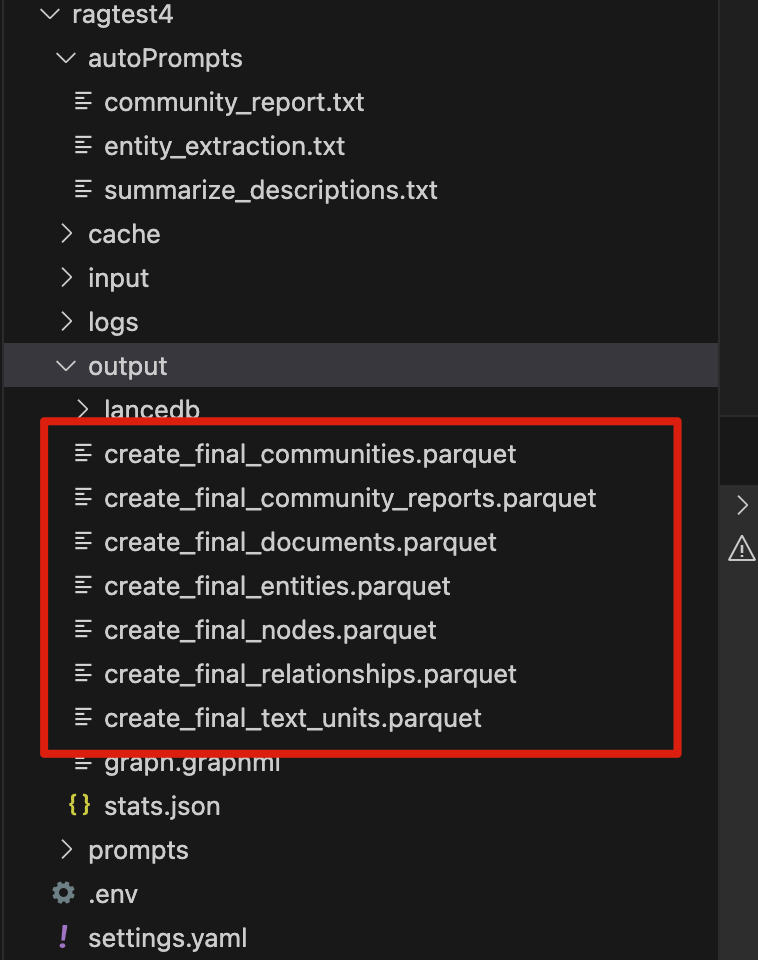
**Step 5: create the graphrag index and it will create lancedb, parquet and graphml files in the output folder automatically and we mainly used parquet files because it has tables and it can be visualized.**

**graphrag index --root ./ragtest**

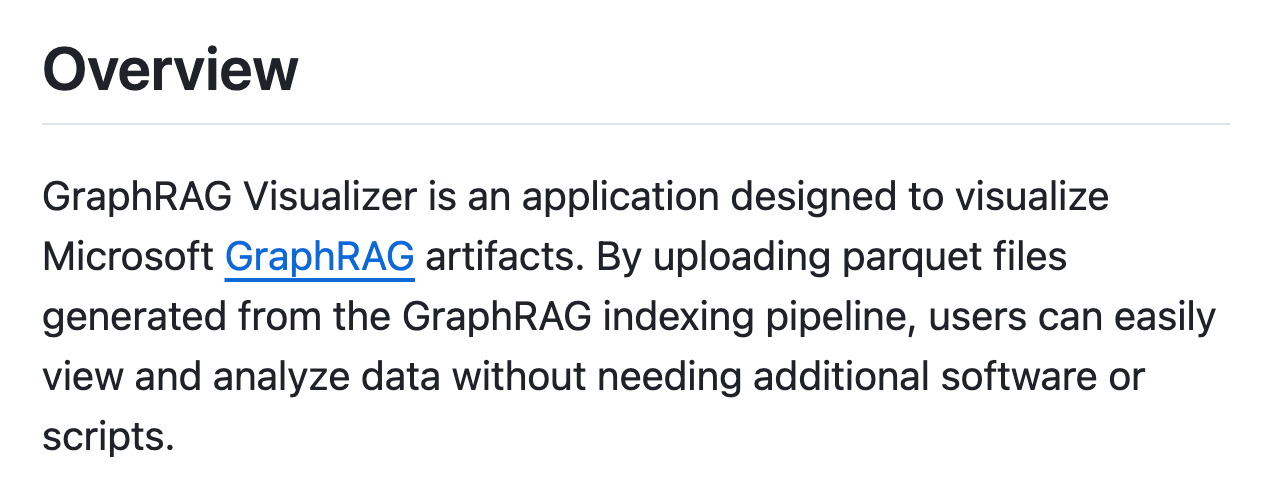
**graphrag query --root ./ragtest**

*If we want to add more .txt file to input folder, update the graph*

**graphrag update --root ./ragtest**



**Step 6: use GraphRAG Visualizer to deal with above parquest files.**



**<https://github.com/noworneverev/graphrag-visualizer>**

