National Research University Higher School of Economics

Faculty of Computer Science
Programme: Data Science and Business Analytics

BACHELOR'S THESIS
Software Project
HSE Smart Assistant: Integrated Voice Assistant for Students

Prepared by the student of Group DSBA202, Year 4 (year of study),
Prokhorova Anastasiia Yurievna

signature

Thesis Supervisor:
Candidate of Sciences* (PhD) in Mathematical Support and Software in Computers, Complexes and Computer Networks, First Deputy Dean on the Faculty of Computer Science, Voznesenskaya Tamara Vasilievna

Co-supervisor:
Visiting Lecturer of the Faculty of Computer Science, Department of Big Data and Information Retrieval, Andreeva Daria Alexandrovna

signature

Moscow
2024

Abstract

This work focuses on developing and analyzing metrics to evaluate the performance of Retrieval-Augmented Generation (RAG) models, particularly in generating accurate and contextually relevant answers to posed questions. The primary objective is to assess how effectively these models retrieve necessary context and generate complete and logical answers. This study encompasses several key stages, beginning with a literature review of current research and methodologies in evaluating machine learning models dealing with text data. Various metrics, including distance metrics (Cosine similarity, Euclidean distance, Jaccard similarity), ranking metrics (MAP@k, MRR, NDCG), and complex metrics (ROUGE, BERTScore), are implemented and tested on six different RAG models. The work highlights significant differences in model performance, showing superior results across multiple metrics. Additionally, ChtaGPT-4 and YandexGPT-3 share their opinions on the obtained results. The model elaborates that despite lexical deviations, but successfully preserving the semantic aspects of reference texts, such models can be viable candidates for further integration into an MVP project for a smart assistant at the National Research University Higher School of Economics. Future work may explore developing more sophisticated metrics and enhancing the model evaluation process.

Keywords: Retrieval-Augmented Generation, RAG models, evaluation metrics, machine learning, natural language processing, cosine similarity, ROUGE, BERTScore, semantic similarity, model performance.

GitHub repository with materials: https://github.com/Anastation67/RAG-diploma

Table of content

1. Introduction

Nowadays technologies are becoming more advanced. People's demands for the capabilities of engineering and technology are growing every time, along with the volumes of information that need to be processed, stored and analyzed. This is especially true in the field of machine learning and artificial intelligence, where new approaches can fundamentally change the way we interact with information and are able to efficiently process big data and generate useful insights.

I was faced with the task of not only studying, but also applying modern research methods in real conditions.

The goal of this work is to develop and analyze the effectiveness of metrics to evaluate the performance of Retrieval-Augmented Generation (RAG) models, which are used to generate answers to posed questions based on relevant context. The main focus of the work is on assessing the accuracy with which the models find the necessary context, as well as assessing the completeness and logic of the formulated answer to the question.

This paper describes the key stages of studying RAG and LLM evaluation metrics, starting with the study of current research works and methodologies in the field of evaluation metrics for machine learning models working with text data and ending with the implementation of different metrics for assessing the quality of RAG models and quality of generated context.

This work is part of a team effort, the goal of which is to create an MVP smart assistant for students at the National Research University Higher School of Economics, where my task is to evaluate the work of the assistant, namely, how accurately the model understands the question, determines and generates an answer to it. In this work we need our assistant to answer correctly and as full as it can, but exact ideal answer that the university’s study office gives.

1.1. Goal and tasks

1.1.1. Goal

The main goal of the diploma is the development and detailed study of metrics to assess the effectiveness of Retrieval-Augmented Generation (RAG) models, which are used to generate answers to questions based on the selected context. The goal is to analyze how accurately models find the desired context and fully answer the questions posed.

1.1.2. Tasks

1. **Study the literature on metrics for evaluating machine learning models:**
   - Familiarity with current scientific publications and methods used to evaluate machine learning models, particularly in the context of natural language processing and information retrieval.
   - Selecting the most relevant and effective metrics for analyzing the performance of RAG models
2. **Implement and integrate metrics into the model evaluation process**:
   - Implementing and testing some key metrics that can evaluate the quality of response generation by RAG models.
3. **Analyze obtained results**:
   - Assessing the compliance of the model results with the requirements of the tasks, identifying its strengths and weaknesses in the context of the accuracy and completeness of the answers.
4. **Analyze possible improvements of the model**

2. Literature Review

Using RAG is a relatively new way of information retrieval in large language models.

This idea was one of the first to be developed by a team at Facebook AI Research and University College London, combining the power of pre-trained language models with information retrieval.

RAG stands for Retrieval-Augmented Generation and is a machine learning technique that combines information extraction and text generation to create more informative and relevant answers. This approach allows the model to use up-to-date data from external sources, making responses more accurate and informative than traditional methods that rely only on pre-trained data. It also provides faster access to the desired context, as opposed to costly additional training of the model on the necessary data, of which there is a huge amount in the industry.

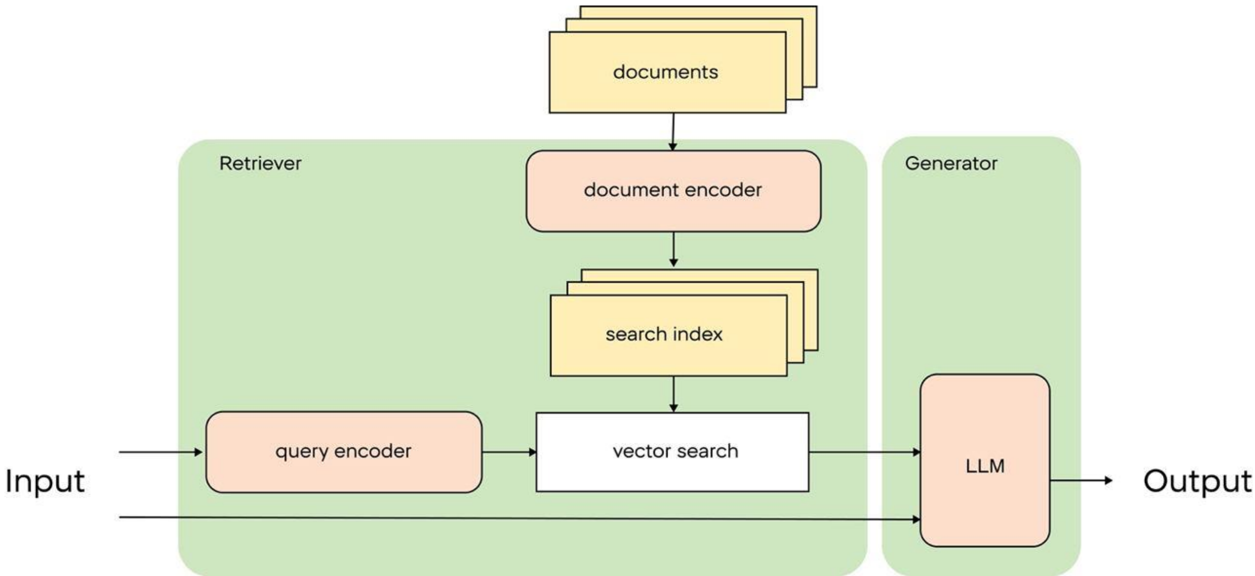

Figure 1: Overview of RAG approach 1

Figure 1 introduces the logic under the hood of RAG methods. The Retrieval stage is responsible for searching and selecting relevant documents or pieces of information from an external database or indexed source. This step helps collect the context that will be used to generate the response. The generator uses the selected data to generate an answer. In this process, the model not only uses its knowledge gained through training, but also integrates information from the extracted data to create more accurate and detailed text. In the next step, the model combines the extracted information with its own knowledge to form a final answer. This is done by optimizing the coherence and relevance of the text in relation to the question or task at hand.

This RAG-based approach can help significantly improve the quality of responses, making them more suitable for complex queries and questions that require up-to-date information.

The article by **Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks"**[1] presents the principles of Retrieval-Augmented Generation. The authors explain how pre-trained language models significantly improve the quality of text generation when information retrieval algorithms are added to them. It explains the importance of carefully selecting context from the database before using it to generate a response.

In this work, the model was evaluated on several datasets for open-domain QA tasks such as Natural Questions, WebQuestions, and CuratedTrec, as well as fact-checking tasks such as FEVER.

The work demonstrates that adding nonparametric memory to pretrained transformer models significantly improves the quality of response generation.

RAG performs better than basic seq2seq models and models specialized for answer extraction. RAG also offers mechanisms to update knowledge more dynamically and ensure interpretability of responses, which is important for real-world applications.

To gain a deeper understanding of the operation of RAG models and become familiar with the metrics for evaluating model results, I decided to start by looking at the book **Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze, "Introduction to Information Retrieval"**[2]

The book provides the theoretical background needed when working with RAGs and evaluating models. The book details the basic principles of information retrieval, including indexing, ranking, and query processing. These concepts are directly related to how RAG models use document databases to generate responses.

One of the book's chapters is devoted to vector models, such as TF-IDF, which are used to represent text documents as vectors. This, in the context of the topic at hand, is directly related to the "retriever" component of the RAG model, which retrieves relevant documents based on the similarity of the query and the document.

Chapter 8, Evaluation in information retrieval, discusses various metrics for evaluating the effectiveness of information retrieval systems and evaluation of unranked retrieval sets, such as precision, recall, and F-measure.

Precision measures the proportion of retrieved documents that are truly relevant. Precision is calculated as the ratio of the number of relevant documents to the total number of documents. In the context of search, this shows how accurately the system selects only relevant documents.

Recall measures the proportion of all relevant documents that are retrieved by the system. It is the ratio of the number of relevant documents retrieved to the total number of relevant documents available in the database.

F-measure is a single measure that trades off precision versus recall, which is the weighted harmonic mean of precision and recall. This metric is used to evaluate the balance between precision and recall, which is especially important when both metrics need to be considered simultaneously. The F-measure achieves a high value only when both precision and recall are high, making it useful for comparing systems that can optimize one metric at the expense of another.

The author also describes evaluation metrics Evaluation of ranked retrieval results, for example, MAP, noting that among evaluation measures, MAP has been shown to have especially good discrimination and stability. This metric is a combination of precision at k (P@k) and average precision (AP) metrics.

P@k is calculated as the proportion of relevant elements among the first K elements in the ranked list, and AP is the average precision for a single query, which is calculated by averaging the P@k values taken after each relevant element in the ranked list. For example, if the relevant elements are at positions 1, 2 and 6, the AP will be the average of P@1, P@2 and P@6.

Precision at k metric has the benefit of not needing an estimate of the number of relevant documents, but it has the drawbacks of being the least stable among frequently used evaluation metrics and of not averaging effectively, as the total number of relevant documents per query significantly impacts precision at k.

A final approach that has seen increasing adoption, especially when employed with machine learning approaches to ranking is measures of cumulative gain, and in particular normalized discounted cumulative gain (NDCG).

Cumulative gain (CG) is the sum of the relevance scores assigned to the results in a result list up to a particular rank position. If all results have the same relevance or if there is no discounting based on position, this would just be the simple sum of the relevance scores. Thus the discounted cumulative gain improves upon CG by introducing a discount factor that reduces the importance of items retrieved at lower ranks. Finally, NDCG normalizes the discounted cumulative gain (DCG) value by the ideal DCG, which is the DCG value obtained by arranging the results in the perfect order of relevance.

NDCG is essential for scenarios where the relevance of items in a result set can vary in degrees and where the order of items matters significantly.

In this book, following subchapter 8.2 Standard test collections, I learned about the existence of Text Retrieval Conferences. After further research and searching for recordings of this event, I learned that TREC hosts question and answer sessions where participants share the results of using RAG. They present data obtained from extensive documents in various fields, analyzing the effectiveness of RAG, indicating where the system worked well and where it was less effective than expected.

The **TREC Question Answering**[3] and Legal Tracks provide practical examples of how information retrieval systems are evaluated in real-world settings. These reviews provide insight into which metrics (e.g., MRR, MAP) and approaches (e.g., response relevance assessment) can be adapted to evaluate RAGs.

At the **2003 TREC conference**[5], participants were asked to answer evidence-based questions using limited length responses (50 or 250 bytes). The systems were required to return a ranked list of five [document-id, answer-string] pairs for each question.

To evaluate the results, the Mean Reciprocal Rank (MRR) metric was used - the average mutual rank score (MRR). For each question, the reciprocal ranking of the first correct answer was calculated. If the correct answer was not found among the five sentences, a rank of 0 was assigned. The average of all questions determined the MRR for the submission. This metric is well suited for comparative evaluation of different approaches.

As part of **Legal Track 2006**[4], participants were asked to find and submit documents that answered queries related to legal issues. This differs from other TREC QA tasks because the emphasis here is on searching and retrieving long-form documents rather than short answers.

The following assessments were used: Relevance Judgments - Assessments of the relevance of documents, which could vary from assessor to assessor, rather than traditional standard answers, as well as classic metrics Precision, Recall, and F1-measure - information search metrics for assessing the accuracy and completeness of results, and also their harmonic mean (F1).

The latter metrics have already been discussed in more detail in this work. But regarding Relevance Judgments, it is worth emphasizing that the use of human assessors to assess the correctness of answers allows a more objective assessment of the model’s performance. Differences in judges' opinions can have a significant impact on how a system's performance is assessed, especially when it comes to qualitative response scores. The adjudication methods used and cross-validation of judges' opinions help to reduce subjective differences and increase the reliability of assessment results.

Such metrics demonstrate different approaches to measuring and comparing the performance of Q&A systems under different scenarios and conditions. This not only helps improve answer-finding technologies, but also helps set the standard for future research in natural language processing.

I found the publications from the conferences ACL (Association for Computational Linguistics), EMNLP (Empirical Methods in Natural Language Processing) and NeurIPS (Neural Information Processing Systems), which cover modern research in the field of NLP, informative.

As part of the 2004 ACL Workshop, I found the method presented in the article **"ROUGE: A Package for Automatic Evaluation of Summaries" by Chin-Yew Lin**[7] interesting. The author introduces metric ROUGE (Recall-Oriented Understudy for Gisting Evaluation) to automatically determine the quality of a summary by comparing it to other (ideal) summaries created by humans. Author introduces four different types of ROUGE metrics: ROUGE-N, ROUGE-L, ROUGE-W, and ROUGE-S and draws an analogy between the **BLUE metric**[6] for assessing the quality of text translation with its own metric.

1. ROUGE-N: N-gram Co-Occurrence Statistics

ROUGE-N measures the n-gram overlap between the candidate and reference reports. This metric is based on n-gram match statistics, and its formula involves calculating the ratio of the number of matching n-grams in the candidate summary to the total number of n-grams in the reference summary. It is a measure of completeness because it evaluates how much of the reference summary was reproduced in the candidate summary.

2. ROUGE-L: Longest Common Subsequence

ROUGE-L uses Largest Common Subsequence (LCS) to evaluate the similarity between two texts. This metric does not require consistent word matches, allowing the quality of summaries to be assessed at a more global and structural level. ROUGE-L is calculated based on the proportion of words that form the LCS between the candidate and reference summaries, reflecting both precision and recall.

3. ROUGE-W: Weighted Longest Common Subsequence

ROUGE-W improves on the ROUGE-L metric by adding weighting to the LCS, allowing it to give more weight to long consecutive match segments. This is done in order to take into account not only the fact of the coincidence of words, but also their sequential arrangement, which can be important for maintaining the semantic integrity of the text.

4. ROUGE-S: Skip-Bigram Co-Occurrence Statistics

ROUGE-S evaluates summaries using skip bigram matching statistics. This allows you to account for pairs of words that may be separated by other words, but still maintain their order in the sentence. This is especially useful for evaluating texts where word order is important but where insertions or changes may occur in the text.

3. Experiments settings

3.1. Preliminary Stage

Due to the fact that the purpose of this work is to directly work with metrics and evaluate the results of the model, as a preparatory stage of my work and main subject area of the work of my colleague there is a need to investigate developed models and provide main description of them.

Overall information

Model 1	*Model 2*	*Model 3*	*Model 4*	*Model 5*	*Model 6*
bert-base-multilingual-cased + Cosine similarity	*cointegrated/rubert-tiny2 + Cosine similarity*	*cointegrated/rubert-tiny2 + Euclidean similarity*	*Jaccard similarity*	*Jaccard similarity + Lemmatization + Query Expansion + YandexGPT assistance*	*cointegrated/rubert-tiny2 + Cosine similarity + Query Expansion + YandexGPT*

Language model: YandexGPT Lite
Language of documents: Russian
Maximum chunks length: 600 tokens
Documentation:

- *Regulations on the organization of intermediate certification and ongoing monitoring of student progress at the National Research University Higher School of Economics (Положение об организации промежуточной аттестации и текущего контроля успеваемости)*

For more information on RAG models and their specifications I would recommend referring to my colleague's work, whose work focuses on developing different Retrieval-Augmented Generation models that are evaluated here.

In order to evaluate models I created a dataset of 61 question and answers based on information provided in the documentation.

Once LLM with RAG and document preparation is done, I was to determine which metrics described above to choose and furtherly implement them.

3.2. Metrics Selection

In general, the evaluation metrics for the LLM model with RAG can be divided into four groups:

1. distance metrics
2. ranking metrics
3. metrics based on pre-trained llm models
4. llm-based metrics: model's reviews

For a more comprehensive assessment, I decided to look at the results of metrics from all three groups.

Distance metrics allow us to evaluate the semantic proximity between the generated text and the true (correct) texts. I have chosen the following four distance metrics.

Cosine similarity measures the cosine of the angle between two vectors in multidimensional space. The value ranges from -1 to 1. Values less than 0 indicate opposite vectors, 0 indicates no similarity between the vectors, and values close to 1 indicate close similarity between the vectors.

Jaccard Similarity also measures the similarity between two data sets. In this case, the metric is calculated as the ratio of the size of the intersection of sets to the size of their union. The value ranges from 0 to 1. Value close to 1 indicate close similarity between two sets, while values close to 0 show that there is no common elements.

Euclidean distance is the usual distance between two points in Euclidean space. In the context of multidimensional spaces, this is the distance between two vectors. The closer the distance to 0 the more answers coincide.

Ranking metrics are used to assess the quality of ordering a list of documents or answers by their relevance. It is worth considering the previously discussed metrics MAP@k, MRR and DCNG. These metrics were discussed in details above.

However, it is worth mentioning that all values of these metrics lie between 0 and 1.

MAP@k interprets as follows: the closer it to 1 the more relevant elements in the first positions are ranked.

MRR interprets as follows: close to 1 means that the first relevant document almost always on the first place

NDCG score ranges from 0 to 1, where 1 means perfect ranking and all relevant documents (chunks) are ranked in the best possible order.

I also implemented metrics that are based on large language models. I used ROUGE and BERTScore.

ROUGE is used to evaluate automatically generated text against a reference text, usually generated by a human. From ROUGE family I will observe ROUGE-N and ROUGE-L metrics: ROUGE-1: evaluates single word matches (1-gram), ROUGE-2: Evaluates the matching of pairs of words (digrams) and ROUGE-L: based on the length of the longest common subsequence.

BERTScore evaluates text quality by comparing semantic similarity between a generated text and an original text at the level of vector representations. The higher the score the more part of the generated answer is semantically similar to the correct answer.

LLM-based metrics refer to results that are provided by other large language models like GPT4 and YandexGPT-3.

4. Results evaluation

After deciding on metrics, let us continue with implementation and further analysis of obtained results.

4.1. Distance Metrics

These metrics evaluate the semantic proximity between the model response and the reference response. For each of 61 questions I have obtained a model's answer, calculated distance metrics for every generated answer and corresponding true answer, and finally, to make results more representative, calculated means for every metric. The results are presented in the table below.

Metric	Description	Model	Result		Interpretation
Cosine similarity	Uses cosine similarity between vectors of words, sentences or documents	M1	**0.103**	10.3 %	The values are 20% and lower, which indicates insignificant similarity between the vectors, meaning that the text of the models' answers slightly coincide in words with the correct answer on average. Although some of the words in the vectors may be common, in general the texts have quite different patterns. Still models 2 and 3 have better results than others
		M2	**0.216**	21.6 %	
		M3	**0.192**	19.2 %	
		M4	**0.171**	17.1 %	
		M5	**0.166**	16.6 %	
		M6	**0.151**	15.1 %	
Euclidian distance	Measures the Euclidean distance between vectors in multidimensional space	M1	**1.337**		The closets value to 1 has provided the model 2, which indicates that there is a noticeable, but not too large distance between the vector representations of texts on average. From the point of view of multidimensional space, this may mean that the texts are not identical, but also do not differ greatly in their characteristics.
		M2	**1.247**		
		M3	**1.268**		
		M4	**1.284**		
		M5	**1.289**		
		M6	**1.301**		
Jaccard similarity	Used to measure similarity and diversity between sets	M1	**0.050**	5 %	Obtained results are extremely poor, not even reaching 20% of texts on average. Such results indicate that on average words used by models in their answers are not coincide with words
		M2	**0.130**	13 %	
		M3	**0.109**	10.9 %	
		M4	**0.107**	10.7 %	
		M5	**0.098**	9.8 %	

		M6	**0.089**	8.9 %	used in true answers. Still the best result among models provided model 3

Table 1: Distance metrics results of all 6 models

Now, I will introduce some examples of outputs of the models. Let me fix the query I am asking the model and the correct answer, that the model ideally should follow.

Query: Что подразумевают под пререквизитами в образовательном процессе НИУ ВШЭ?

Sufficient answer: Пререквизиты в образовательных программах НИУ ВШЭ подразумевают набор дисциплин и компетенций, которые студент должен освоить перед началом обучения по конкретной дисциплине. Эти дисциплины называются дисциплинами-пререквизитами и обеспечивают необходимую базу знаний для успешного усвоения новых материалов.

Model 1 response: В предоставленных вами документах нет прямого ответа на этот вопрос. Согласно первому документу, пререквизиты — это отдельные Дисциплины, по которым у студента должна быть академическая задолженность в случае, если он не выбрал тему курсовой работы. Второй документ предлагает следующую информацию: Пререквизиты — знания и навыки, необходимые для успешного освоения дисциплины. Они позволяют студенту лучше понять материал и успешно выполнить задания. Например, пререквизитом для изучения математического анализа может быть знание основ математики.Из контекста можно сделать вывод, что в НИУ ВШЭ пререквизиты означают требования к предыдущему обучению, необходимые для понимания и успешного освоения новой Дисциплины.

Cosine similarity	0.135 (13.5 %)	vectors are orthogonal, no similarity
Euclidean distance	1,315	the straight-line distance between two points in Euclidean space, that suggests that the vectors are moderately far apart in this space
Jaccard similarity	0.075 (7.5%)	a very low degree of similarity between the sets

Table 2: Example 1

Model 2 response: Пререквизиты — это перечень результатов обучения или компетенций, которыми должен обладать студент, а также список учебных дисциплин, которые студент должен пройти перед началом изучения определённой учебной дисциплины или элемента образовательной программы.Пререквизиты нужны для того, чтобы обеспечить логическую последовательность и преемственность в образовательном процессе, а также для того, чтобы студент получил необходимые знания и навыки для успешного изучения последующих дисциплин.Например, студент, который изучает дисциплину «Математика 1», должен иметь базовые знания и навыки по математике и уметь выполнять основные математические операции. Но для того, чтобы он смог изучить дисциплину «Статистика 1», ему необходимо предварительно освоить дисциплину «Информатика 1», которая содержит в себе основы работы с компьютером и программным обеспечением. Поэтому «Информатика 1» становится пререквизитом для «Статистики 1».Блокирующий элемент контроля — это элемент контроля, который является обязательным для изучения дисциплины и неудовлетворительная оценка за который приравнивается к неудовлетворительной оценке за дисциплину в целом.Если студент не сдал блокирующий элемент контроля или не смог выполнить его пересдачу, он не сможет получить допуск к экзамену или зачёту по дисциплине.В приведённом вами отрывке из документов НИУ ВШЭ говорится о том, что блокирующим элементом контроля по дисциплине «Совместная программа ВШЭ и РЭШ. Экономика 1. Микроэкономика 1.1» является «Экономика 1.0. Часть 1. Основы экономической теории», а также о том, как происходит пересдача блокирующих элементов контроля.Вы можете использовать эту информацию для того, чтобы лучше понять требования к изучению дисциплины и подготовиться к пересдаче блокирующего элемента.		
Cosine similarity	0.202 (20.2 %)	twice higher than model 1 showed, still almost no similarity
Euclidean distance	1,263	the vectors are moderately far apart in the space
Jaccard similarity	0.098 (9.8%)	still a low degree of similarity between the sets

Table 3: Example 1 continue

These results show that both answers are quite dissimilar to answer. Despite the low similarity metrics, Model 2 provides a more detailed response that covers more aspects of the sufficient answer, indicating a deeper understanding, although still not closely aligned with the ideal answer.

Overall, the texts have certain basic similarities, but also significant differences in structure and content. Based on distance metrics observed, all models generally perform poorly in terms of similarity to the correct answers. However, Model 2 consistently shows better performance across different metrics.

While distance metrics like provide additional insights into the models' performance, they should be considered while evaluating RAG performance. It is better to choose a combination of metrics like Cosine similarity and Jaccard similarity, or calculating Euclidean distance and Jaccard similarity as it provides more global result assessment.

4.2. Ranking Metrics

Such metrics are used to evaluate the quality of ordering a list of documents or answers by their relevance. As mentioned earlier, I decided to implement the MAP@k, MRR and NDCG metrics.

Metrics	Description	Model	Result		Interpretation
MAP@K	Mean Average Precision	M1	**0.160**	16 %	Model 1 shows the best provided results. On average, M1 is more precise in predicting relevant items at rank K compared to the other models. Models 2, 3, and 6 performs almost as well as model 1. Model 4 precision is significantly lower than the other models.
		M2	**0.146**	14.6 %	
		M3	**0.146**	14.6 %	
		M4	**0.031**	3.1 %	
		M5	**0.064**	6.4 %	
		M6	**0.144**	14.4 %	
NDCG	Normalized Discounted Cumulative Gain	M1	**0.605**	60.5 %	Again, model 1 is the most effective in ranking relevant item. It significantly outperforms other models. Models 2, 3, and 6 are showing middle results. Model 4 fails to rank relevant items properly.
		M2	**0.436**	43.6 %	
		M3	**0.436**	43.6 %	
		M4	**0.000**	0 %	
		M5	**0.256**	25.6 %	
		M6	**0.429**	42.9 %	
MRR	Mean Reciprocal Rank	M1	**0.339**	33.9 %	Model 6 has the highest MRR, indicating it is the most effective in ranking the most relevant items at the top positions. Models 2,3, and 1 have moderate results, while model 4 again shows the poorest results.
		M2	**0.377**	37.7 %	
		M3	**0.377**	37.7 %	
		M4	**0.060**	6 %	
		M5	**0.176**	17.6 %	
		M6	**0.572**	57.2 %	

Table 4: Ranking metrics results of all 6 models

Overall, the best results showed models 1 and 6. However, model 1 had only been once on top showing the best NDCG, while other times the model indicated low results among others. So the best performance from documents ranking perspective showed the sixth model, indicating its strong ability to rank relevant items at the top positions.

Let me illustrate an example.

> Take the same query, the same sufficient answer and the same generated model 1 and model 2 answers. An average precision at top 3 relevant document metric (**AP@3**) appeared to be equal to **0.167** for model 1 and **0.667** for model 2, indicating that model 1 found 1 relevant document and model 2 found two relevance.
>
> Indeed, actual **chunk** that are to be found are **[1, 2, 15]**, while the models' developed answers based on other chunks: model 1 chunks **[22, 2, 10]** and model 2 chunks are **[1, 2, 38].** Texts of each chunk for this example see in repository in the document "baseline chunks". Consequently, relevance-based

metrics like **Mean Reciprocal Rank** directly depend on determined chunks: model 1 **MRR = 0.5** and model 2 **MRR = 1**. The results of Model 2 are indeed better, which means more efficient extraction of relevant information. This is a fairly good indicator/metric for assessing RAG, which is definitely worth taking into account when searching for the best model. High values of AP@3 and MRR confirm that Model 2 performs the task of finding relevant information better than Model 1. Therefore, these metrics are important when comparing and selecting the most effective model.

Table 5: Example 2

4.3. Complex Metrics

Metric		Description	Model	Result		Interpretation
ROUGE-1	Recall	the ratio of the number of single words from the reference text that also occur in the generated text to the total number of words in the reference text	M1	**0.187**	18.7 %	At most 42% of words from the reference texts appeared in the second model texts. Other models seem to have moderate results comparing with the second model.
			M2	**0.420**	42 %	
			M3	**0.322**	32.2 %	
			M4	**0.293**	29.3 %	
			M5	**0.273**	27.3 %	
			M6	**0.275**	27.5 %	
	Precision	the ratio of the number of single words from the generated text that match the words in the reference text to the total number of words in the generated text	M1	**0.053**	5.3 %	Only at most 17.3% of the words from the second model texts are presented in the reference texts. First model has the poorest results.
			M2	**0.173**	17.3 %	
			M3	**0.146**	14.6 %	
			M4	**0.144**	14.4 %	
			M5	**0.126**	12.6 %	
			M6	**0.116**	11.6 %	
	F1-score	the harmonic mean between recall and precision for 1-gram	M1	**0.073**	7.3 %	Overall, extremely low F1 score, indicating low match, still the best among provided results are obtained by the second model, with the first model in loosers.
			M2	**0.219**	21.9 %	
			M3	**0.193**	19.3 %	
			M4	**0.183**	18.3 %	
			M5	**0.170**	17 %	
			M6	**0.155**	15.5 %	
ROUGE-2	Recall	the ratio of the number of matching bigrams to the total number of bigrams in the standard answer	M1	**0.064**	6.4 %	very low completeness, indicates that very few bigrams from the standard were found in the model text. Second model shows better results, while the first model is the poorest here
			M2	**0.194**	19.4 %	
			M3	**0.123**	12.3 %	
			M4	**0.102**	10.2 %	
			M5	**0.104**	10.4 %	
			M6	**0.094**	9.4 %	

	Precision	the ratio of the number of matching bigrams to the total number of bigrams in the generated response	M1	**0.124**	12.4 %	indicates very low accuracy, most of the bigrams generated by the models do not match the standard
			M2	**0.067**	6.7 %	
			M3	**0.052**	5.2 %	
			M4	**0.052**	5.2 %	
			M5	**0.041**	4.1 %	
			M6	**0.037**	3.7 %	
	F1-score	the harmonic mean between recall and precision for bigrams	M1	**0.019**	1.9 %	very low value, which indicates that from the point of view of bigrams there is very little overlap between texts
			M2	**0.087**	8.7 %	
			M3	**0.070**	7 %	
			M4	**0.064**	6.4 %	
			M5	**0.058**	5.8 %	
			M6	**0.050**	5 %	
ROUGE-L	Recall	the proportion of the longest common subsequence of words from the reference text reproduced in the generated text	M1	**0.177**	17.7 %	about 40% of the longest sequence of words from the standard occurs in the second model texts, which is the highest results. Others results are moderate
			M2	**0.399**	39.9 %	
			M3	**0.304**	30.4 %	
			M4	**0.273**	27.3 %	
			M5	**0.254**	25.4 %	
			M6	**0.258**	25.8 %	
	Precision	the proportion of words from the longest common subsequence of the generated text that matches the reference text	M1	**0.051**	5.1 %	approximately 16% of the longest sequence of words from the second model texts is presented in the true answers. All results seem to be almost at the same level
			M2	**0.163**	16.3 %	
			M3	**0.138**	13.8 %	
			M4	**0.134**	13.4 %	
			M5	**0.117**	11.7 %	
			M6	**0.109**	10.9 %	
	F1-score	the harmonic mean between recall and precision	M1	**0.068**	6.8 %	given the low precision and relatively moderate recall, the F-measure seems to be at model 20.7 % and at least 6.8%. Second model wins in here.
			M2	**0.207**	20.7 %	
			M3	**0.182**	18.2 %	
			M4	**0.171**	17. 1 %	
			M5	**0.159**	15.9 %	
			M6	**0.146**	14.6 %	

Bert score	Recall	the proportion of semantically relevant words in the generated text that correspond to words in the reference text	M1	**0.657**	65.7 %	About 77% of the words in the generated texts have a high degree of semantic similarity to the words in the reference texts. This indicates good accuracy, showing that most of the words in the models' responses have a semantic match to the words in the references. All models show good results, still the best was obtained the second model.
			M2	**0.770**	77 %	
			M3	**0.730**	73 %	
			M4	**0.712**	71.2 %	
			M5	**0.709**	70.9 %	
			M6	**0.706**	70.6 %	
	Precision	the proportion of words in the reference text for which semantically corresponding words were found in the generated text	M1	**0.606**	60.6 %	approximately 69% of words from the reference texts found their matches in the model texts. This is a fairly high precision, which suggests that the model captures the content of the reference texts well. The leader is still the second model.
			M2	**0.685**	68.5 %	
			M3	**0.672**	67.2 %	
			M4	**0.668**	66.8 %	
			M5	**0.663**	66.3 %	
			M6	**0.660**	66 %	
	F1-score	the harmonic mean between recall and precision	M1	**0.628**	62.8 %	The value is a balance between precision and recall. This shows that the model copes quite well with the tasks of reproducing the standard texts, paying attention to both the quality of the generated words and their correspondence with the standards.
			M2	**0.724**	72.4 %	
			M3	**0.699**	69.9 %	
			M4	**0.689**	68.9 %	
			M5	**0.684**	68.4 %	
			M6	**0.681**	68.1 %	

Table 6: Complex metrics results of all 6 models

ROUGE show that the generated text differs significantly from the reference text. The metrics indicate low co-occurrence at both the single word level and the bigram and long sequence level. This may indicate that the generated text either does not accurately reflect the information from the reference text, or represents it in a completely different way. Still, the best results has showed the second model.

At the same time, BERTScore indicates relatively good quality of the generated texts compared to the reference texts, with some advantage in completeness over accuracy. A high level of recall and a fairly good F1 Score indicate that the model effectively reproduces information from the reference text while maintaining semantic similarity. All tested models showed relatively good semantic similarity with reference answers.

Overall, the discrepancy between the ROUGE and BERTScore results may be due to differences in what these metrics measure:

- ROUGE focuses on lexical co-occurrence by assessing the presence of exact words and their sequences.
- BERTScore takes into account semantic similarity, perhaps allowing the model to move away from precise wording but maintaining overall meaning and context.

In context of this work I mostly need from a model to have good overall meaning and context, probably giving preferences to BERTScore results.

The example discussed earlier showed ROUGE-L results to be weak, F1 score is around 13-15%. While BERTScore results are pleasantly surprising (F1 score is 72%), showing that both true answer, model 1 and model 2 responses are semantically close. When a person reads and compares the two answers, it is evident that both model responses correctly address the question. However, they include additional details not explicitly requested in the query. Despite its low score, ROUGE-L might not fully capture the quality of the responses, especially when they are semantically correct but differ in wording or structure. High BERTScore indicates that the responses are semantically close to the ideal answer, which aligns with human judgment.

Table 7: Example 3

4.4. LLM-based metrics: models' reviews

Let us have a look at how other large language models assess the result of implemented models.

I asked GPT-4o to assess completeness, logic in formulating the answer, and fullness of the answers generated to 61 questions from each of the six models.

My query was as follows: You have 6 Excel files, each of which has a 'Model response' column - it contains the answers of each of the six models to the corresponding questions from the 'Query' column. I need you to compare the overall performance of these models based on their responses. It is necessary to evaluate the consistency of the answers, adequacy, accuracy, logic and completeness of the answers, evaluate each model on a scale from 1 to 100, where 100 is the ideal complete, understandable and logical answer.

GPT-4 answer:

Model	Accuracy Score	Consistency Score	Logic Score	Completeness Score	Overall Score
Model 1	74.70	73.64	76.52	76.21	75.27
Model 2	75.16	74.04	72.31	73.81	73.83
Model 3	75.24	77.55	74.84	76.07	75.93
Model 4	74.71	74.66	73.77	75.66	74.70
Model 5	74.78	74.56	74.16	74.93	74.61
Model 6	72.95	75.42	74.67	74.04	74.27

Table 8: Evaluation of models' answers from ChatGPT-4o

Overall, one of the most famous large language model ChatGPT has determined third model to perform better. However, overall scores showing not a significant differences between six models. According to the table 4, model 1 seems to be have the most logic answers, model 2 is the leader in accuracy, model 3 leads consistency score. Other models' results are in the middle.

Using the same practice for the considered examples of models 1 and 2, I asked YandexGPT-3 and chat GPT-4o to evaluate their answers regarding the correct answer using the same criteria. Below are the results.

My query was as follows: I need you to compare responses on query of models 1 and 2. It is necessary to evaluate the consistency of the answers, adequacy, accuracy, logic of formulating the answers and completeness of the answers, evaluate each model on a scale from 1 to 100, where 100 is the ideal answer.

GPT-4 answer:

Criteria	Model 1 Score	Model 2 Score
Consistency	70	85
Adequacy	60	80
Accuracy	65	85
Logic	70	80
Completeness	60	90
Overall Score	**65**	**84**

Table 9 ChatGPT-4o evaluation of example results

Based on these scores, Model 2 provides a more consistent, adequate, accurate, logical, and complete answer compared to Model 1.

YandexGPT-3 answer:

Criterion	Model 1	Model 2
Consistency	80	90
Adequacy	90	95
Accuracy	95	100
Logic of formulation	90	95
Completeness	90	95
Overall score	88	93

Table 10 YandexGPT-3 evaluation of example results

The results show that both models provide relevant and accurate information about the prerequisites in the educational process of the Higher School of Economics (HSE).

Model 1 is more consistent and provides a comprehensive overview of the prerequisites, but it may not be as detailed as Model 2. Model 2 is more detailed and provides a more comprehensive understanding of the prerequisites.

Overall, Model 2 received a slightly higher score, indicating that it provides a more complete and accurate answer to the question about the prerequisites in the HSE educational process.

Table 11 Example 3

Both evaluations indicate that Model 2 outperforms Model 1 across various criteria, providing more detailed, accurate, and comprehensive answers. Both GPT-4 and YandexGPT-3 evaluations agree on the superior performance of Model 2, making it the preferred choice for retrieving relevant and precise information.

4.5. Performance of the models

It is time to understand what might be the reasons of such results of the models.

Surely the model architecture is extremely important as well as the data that the RAG model bases its answers on. RuBert architecture is fine-tuned on large corpora of relevant data which allows RuBerta-based models perform well.

Results could have been more accurate with a greater document base as for now all models are "taught" on just one document *"Regulations on the organization of intermediate certification and ongoing monitoring of student progress at the National Research University Higher School of Economics".*

Different metrics have varying sensitivity to specific aspects of the generated text. Distance metrics measure the semantic similarity between generated text and the reference text, while lexical metrics like ROUGE mostly focuses on exact word matching. So the answer may be semantically correct, but formulated in other words than used in true answer.

At the same time, the best and quite good in general results of ranking metrics showed the last model 6, that is actually an upgrade of the second model. The model indicates its capability to prioritize relevant information accurately. This could be due to adding query expansion step, which resulted in prioritizing more relevant contexts in the retrieval process.

One more factor that surely influences metrics' results is the fact of llm answers generation. In distance metrics or in lexical metrics like ROUGE the way generated answer is presented is important. As large language model formulates the answer based on the context provided every launch differently, this may influence the results as well.

Summing up, the architecture, training data, metric sensitivity, query expansion, and the inherent variability in LLM-generated answers collectively explain the most of observed results of the models.

4.6. Summary of the results

In the table below all results are showed for better representation. Again from table 4 one can see that model 2 is slightly but noticeably different from the results of other models, especially in cosine similarity, Recall BERTScore and consequently F1 score, and ROUGE-1 and ROUGE-L f-scores.

	Model 1	**Model 2**	**Model 3**	**Model 4**
Cosine similarity	0.103	0.216	0.192	0.171
Euclidean distance	1.337	1.247	1.268	1.284
Jaccard similarity	0.050	0.130	0.109	0.107
MAP@k	0.160	0.146	0.146	0.031
MRR	0.339	0.377	0.377	0.060
NDCG	0.605	0.436	0.436	0.000
Rouge (f-score)	**Rouge-1:** 0.073 **Rouge-2:** 0.019 **Rouge-L:** 0.068	**Rouge-1:** 0.219 **Rouge-2:** 0.087 **Rouge-L:** 0.207	**Rouge-1:** 0.193 **Rouge-2:** 0.070 **Rouge-L:** 0.182	**Rouge-1:** 0.183 **Rouge-2:** 0.064 **Rouge-L:** 0.171
BERTScore	**Precision:** 0.606 **Recall:** 0.657 **F1:** 0.628	**Precision:** 0.685 **Recall:** 0.770 **F1:** 0.724	**Precision:** 0.672 **Recall:** 0.730 **F1:** 0.699	**Precision:** 0.668 **Recall:** 0.712 **F1:** 0.689

Table 12: Comparison of all results of Models 1-4

While tables 5 and 6 are demonstrating results of 5 generations of models 5 and 6. The difference between these models and models 1-4 is in the fact that closest chucks of models 1-4 are fixed no matter how many times to ask models the same question whereas the last two models are based on two steps: firstly, the model takes a query, obtains an answer from the YandexGPT and only after that query, the llm answer, and the context are given to the final step of answer generation. Because the model heavily depends on what YandexGPT answers at step one, closest chunks are always different. That is why I decided to measure the performance of these models by fully assessing launch of the models 5 times and take the mean results.

	Model 5: Jaccard search with lemmatization and LLM-based answers					Summary
	Test 1	Test 2	Test 3	Test 4	Test 5	
Cosine similarity	0.176	0.170	0.156	0.155	0.174	0.166
Euclidean distance	1.280	1.286	1.297	1.298	1.282	1.289
Jaccard similarity	0.105	0.104	0.089	0.091	0.100	0.098
MAP@k	0.086	0.066	0.058	0.053	0.059	0.064
MRR	0.219	0.186	0.164	0.148	0.161	0.176
NDCG	0.121	0.000	0.587	0.286	0.286	0.256
Rouge (f-score)	**Rouge-1:** 0.183 **Rouge-2:** 0.065 **Rouge-L:** 0.172	**Rouge-1:** 0.182 **Rouge-2:** 0.062 **Rouge-L:** 0.171	**Rouge-1:** 0.159 **Rouge-2:** 0.050 **Rouge-L:** 0.150	**Rouge-1:** 0.164 **Rouge-2:** 0.057 **Rouge-L:** 0.152	**Rouge-1:** 0.164 **Rouge-2:** 0.057 **Rouge-L:** 0.152	**Rouge-1:** 0.170 **Rouge-2:** 0.058 **Rouge-L:** 0.159
BERTScore	**Precision:** 0.664 **Recall:** 0.708 **F1:** 0.684	**Precision:** 0.669 **Recall:** 0.710 **F1:** 0.688	**Precision:** 0.654 **Recall:** 0.712 **F1:** 0.681	**Precision:** 0.664 **Recall:** 0.708 **F1:** 0.684	**Precision:** 0.664 **Recall:** 0.708 **F1:** 0.684	**Precision:** 0.663 **Recall:** 0.709 **F1:** 0.684

Table 13: Comparison of results of 5 generations of Model 5

	Model 6: RuBert with Cosine search and LLM-based answers					Summary
	Test 1	Test 2	Test 3	Test 4	Test 5	
Cosine similarity	0.174	0.136	0.131	0.159	0.155	0.151
Euclidean distance	1.283	1.313	1.316	1.295	1.299	1.301
Jaccard similarity	0.107	0.080	0.074	0.095	0.088	0.089
MAP@k	0.000	0.000	0.219	0.250	0.249	0.144
MRR	0.533	0.519	0.582	0.596	0.631	0.572
NDCG	0.409	0.447	0.315	0.564	0.409	0.429
Rouge (f-score)	**Rouge-1:** 0.186 **Rouge-2:** 0.063 **Rouge-L:** 0.173	**Rouge-1:** 0.138 **Rouge-2:** 0.040 **Rouge-L:** 0.131	**Rouge-1:** 0.131 **Rouge-2:** 0.041 **Rouge-L:** 0.123	**Rouge-1:** 0.166 **Rouge-2:** 0.058 **Rouge-L:** 0.155	**Rouge-1:** 0.156 **Rouge-2:** 0.050 **Rouge-L:** 0.148	**Rouge-1:** 0.155 **Rouge-2:** 0.050 **Rouge-L:** 0.146
BERTScore	**Precision:** 0.671 **Recall:** 0.728 **F1:** 0.698	**Precision:** 0.656 **Recall:** 0.697 **F1:** 0.675	**Precision:** 0.649 **Recall:** 0.683 **F1:** 0.665	**Precision:** 0.666 **Recall:** 0.712 **F1:** 0.687	**Precision:** 0.656 **Recall:** 0.710 **F1:** 0.681	**Precision:** 0.660 **Recall:** 0.706 **F1:** 0.681

Table 14: Comparison of results of 5 generations of Model 6

While classic distance and ROUGE metrics indicate lexical and structural differences between texts, BERTScore confirms that the generated texts retain the overall meaning and context of the references. In the context of this work and obtained results, model 2 seems to be one of the best models especially in distance and complex metrics results. Metrics based on LLMs also determine the model as the model providing accurate, consistent, and complete answers. This suggests that, despite lexical differences, the model successfully conveys key semantic aspects of the reference text.

Thus, as the goal is to preserve and convey the meaning and context of information, the model can be considered to work satisfactorily, even despite some lexical deviations from the reference text.

As a result of this work, I have assessed six different retrieval-augmented generation models by comparing distance, ranking, and complex metrics, supported by large language model from OpenAI. The final decision on the preferred model to use in the MVP project of HSE voice assistant is the second model of combination of RuBert with cosine similarity vector search.

5. Conclusion

Evaluating RAG models is complex due to the diversity of aspects that need to be considered, such as semantic accuracy, relevance, and completeness.

Metrics like Average Precision at Top 3 (AP@3) and Mean Reciprocal Rank (MRR) are essential for evaluating RAG models as they directly measure the model's ability to retrieve and rank relevant documents accurately.

High values in these metrics, as seen with Model 2's AP@3 (0.667) and MRR (1), indicate effective retrieval of relevant information. These metrics should be heavily weighted in decision-making processes for selecting RAG models.

Semantic similarity is also vital. BERTScore is also highly valuable for evaluating the quality of generated responses. Despite lexical differences, a high BERTScore demonstrates that the generated responses maintain the meaning and context of the reference text. BERTScore aligns well with human judgment, thus it a preferred metric over purely lexical metrics like ROUGE, especially when the goal is to preserve the overall meaning rather than exact word matching. ROUGE metrics may be used as a supplementary metric: it’s useful for evaluating sequence overlap but should not be the sole determinant.

It is important to combine different metrics in order to study the model results more fully and comprehensively. Distance metrics and ROUGE metrics should complement relevance-based and semantic metrics, as well as llm-based metrics.

The evaluations showed that Model 2 performed significantly better in retrieving and presenting relevant information.

6. Further improvement

The authors of the book "**Introduction to Information Retrieval**"[2] talk about text classification and clustering, machine translation, and semantic proximity determination, which can be useful for the further development and evaluation of RAG models.

The further work might be focused on the following:

- develop other more sophisticated metrics like RAGAS and other libraries
- research deeper the problem of creating a 'universal' metric to evaluate any or most RAG models
- try to combine several RAG techniques adding re-ranking step and more documents
- find a combination of model 2 and model 6 advantages and implement it in one model
- work on different prompts of the models and evaluate results
- try non-binary relevance while calculating ranking metrics
- test what value of k is the best for ranking metrics like AP@k, MAP@k etc. in the context of current topic's documentations

7. References

1. Lewis P. et al. Retrieval-augmented generation for knowledge-intensive nlp tasks //Advances in Neural Information Processing Systems. – 2020. – T. 33. – C. 9459-9474.

2. Manning C. D., Raghavan P., Schütze H. Introduction to information retrieval. – Cambridge university press, 2008.

3. Voorhees E. M. et al. The TREC-8 Question Answering Track Evaluation //TREC. – 1999. – T. 1999. – C. 82.

4. Baron J. R., Lewis D. D., Oard D. W. TREC 2006 Legal Track Overview //TREC. – 2006.

5. Voorhees E. M., Buckland L. Overview of the TREC 2003 Question Answering Track //TREC. – 2003. – T. 2003. – C. 54-68.

6. Papineni K. et al. Bleu: a method for automatic evaluation of machine translation //Proceedings of the 40th annual meeting of the Association for Computational Linguistics. – 2002. – C. 311-318.

7. Lin C. Y. Rouge: A package for automatic evaluation of summaries //Text summarization branches out. – 2004. – C. 74-81.