AI-based Text Generation for Semantic Search Robustness : Application to Defence

Claude Fendzi
Airbus Defence and Space
Elancourt , France
claude.fendzi@airbus.com

Bruno Carron
Airbus Defence and Space
Elancourt , France
bruno.carron@airbus.com

Guillaume Gadek
Airbus Defence and Space
Elancourt , France
guillaume.gadek@airbus.com

Abstract—**One of the biggest challenges in successfully applying Artificial Intelligence (AI) in the Defense sector is the availability of trustful domain specific data to train AI models on. These data have to be generated and collected from the real world or acquired through realistic scenarios simulations and validated by operation specialists or domain experts. In real world applications, most of the time these data are classified and difficult to access. Then only a handful coming either from unclassified documents or simulation / realistic scenarios can be made available. In this article, we discuss how Generative AI can be used to generate intelligence-oriented textual data that are semantically similar to a "ground truth" database. The methodology is applied in the frame of the EDIDP AI4DEF project, focusing on one of the use cases, Request for Information (RFI) semantic similarity detection in a database. We expose how a limited corpus has been enriched with noisy AI-generated data. The performances and the robustness of the AI model have been monitored to be kept similar before and after the data augmentation, while a human-in-the loop qualifies the AI-generated data.**

Keywords—Natural Language Processing, data generation, Large Language Models, Request for Information, Robustness, Semantic Similarity Search

I. Introduction

Natural Language Processing (NLP) fields have significantly evolved in the last couple of years, and gained a substantial paradigm shift with the advent of Large Language Models (LLM) [1], [2], [3]. These models, distinguished by their considerable size and comprehensive training data, have demonstrated extraordinary abilities in comprehending and producing human-like text [4], [5]. Recently, the transformer architecture was adopted by the GPT (Generative Pre-trained Transformer) series of models developed by OpenAI [5], including GPT-1 [6], GPT-2 [7] and GPT-3 [8]. These models were pre-trained on a large corpus of text from the internet, and then fine-tuned on specific tasks [9]. They demonstrated the ability to generate coherent and contextually relevant texts, marking a significant step forward in the field, beating the traditional NLP architectures (Recurrent Neural Networks, Long Short-Term Memory, Bidirectional Encoder Representation from Transformers), on various traditional tasks such as text processing or text understanding [10]. Traditional architectures were limited in their ability to model long-range dependencies in text, which is crucial for maintaining coherence over longer passages [11]. These new generations of transformer architectures achieved state-of-the-art results in NLP tasks on publicly available internet data, but to the best of our knowledge, have never been applied in the defence domain. Indeed, in the defence domain, and specifically in the intelligence domain, the data availability is a critical issue, due to confidentiality, security and classification. Thus, generating realistic intelligence domain-specific texts, relying on a generic model that was trained on open Internet data can be a tedious task, as the vocabulary and syntax in the intelligence domain are very specific, and may lead to the out-of-vocabulary (OOV) problem.

In this paper, we propose an approach to leverage large language models (LLMs) to generate intelligence domain specific textual data, keeping an *intelligence analyst* in the validation loop. These generated textual data considered as noise are then added in a reference intelligence text database. The approach is tested on a semantic similarity search task, and we compare the performance of our algorithms before and after the data augmentation. A focus is also made on the robustness to typos and misspelling in the search queries.

"RFI similarity search": comparing texts with respect to their semantics.

A very specific operational task in military headquarters consists in receiving RFIs (request for information)[1] that may bear on *any* topic relative to the current or future area of operations. RFIs are typically questions or notifications to ask either for information elements (whose complexity may range from the coordinates of a sensitive location, to the current security situation evolution perspectives between two ethnic groups), or to begin to be notified about topics. It is common to observe trends in the reception of RFIs, which are often bearing on similar topics or areas. Past RFIs, and their answers, are very likely to provide relevant answer elements; this is however not obvious to successfully retrieve these past RFIs without the dedicated support of the information system itself. We propose to perform this task with the help of an AI.

[1] https://www.intelligence101.com/an-introduction-to-the-intelligence-cycle/

In this paper, we propose a candidate approach to tackle the problem of lack of data in RFI, especially in the defence domain. We bootstrap from a limited number of reference sample data to generate new ones that are semantically similar. In details, our contributions are the following:

- we introduce a specific challenge of NLP: RFI semantic similarity detection.
- we propose and implement a methodology to tackle the lack of available real annotated data in the text modality.
- we propose a method to evaluate the generated data quality, benefiting from the expertise of qualified *human operators.*
- we assess the robustness of the data augmentation process to the RFI semantic similarity detection task.

The rest of the paper is structured as follows: Section II formalizes the operational problem of RFI semantic similarity search and the methodology that has been implemented. Section III describes the datasets and the experimental setups. The results of these experiments are discussed in sections IV and V; Section VI concludes this work.

II. Semantic Similarity Search For Defence

A. *RFI Similarity Search: Operational Requirements and Hypotheses*

The RFI similarity detection activity describes a task performed by an *intelligence analyst*, who receives a Request for Information (RFI) from a *Commander* and tries to find if there exists a similar RFI within a Database (typically, the Intelligence Collection Plan database). From a purely technical point of view, the system relies on an AI algorithm that implements a semantic similarity search engine that allows to compute the RFI similarity in a RFI database. The algorithm supports the *intelligence analyst* in their task: upon creation, the new RFI will automatically be associated with suggestions of past RFIs, along with their answers.

- **1st step:** considering a “first” RFI semantic search query, the system is expected to highlight similar RFIs existing in the database along with a confidence level.
- **2nd step:** the analyst then takes advantage of the retrieved RFIs to better improve their RFI request. This helps them to create more accurate RFIs and / or to collect RFI products which best match the original RFI from the *Commander*. This leads to both time saving and accuracy gain in search results.

Operational constraints and requirements- of the Intelligence Collection Plan Database

The *intelligence analyst* receives the *Commander's* needs and has to write the RFI. The main operational requirements are:

- the needs expressed by the *authority (Commander)* shall be properly taken into account in the RFI request form,
- the RFI query shall fill at least mandatory fields (RFI subject or title, RFI Details, RFI Date information, Areas or geographical zones),
- the system shall not duplicate an existing RFI,
- the system shall create the RFI in the shortest possible time.

Hypotheses

We consider the following hypotheses to support the experiment:

- as a prerequisite, there exists a RFI database (*ground truth*) that supports the search query and gets semantically similar RFI if any.
- the RFI database contains all the information enabling the search,
- the RFI database is realistic and semantically correct to lead to accurate and robust search results,
- the RFI database has been validated by an operation specialist *(intelligence analyst*)

B. *Methodology to assess the robustness to noise*

This study examines how an intelligence domain specific AI-based text-generator can be used to augment or enrich a *ground truth* database which supports the RFI semantic similarity search. The experiment consists in semantically indexing the “manually” validated database in the training step, and performing a semantic search query to retrieve semantically similar RFIs in the inference step. This process is first executed on the ground truth RFI database. Then, the database is augmented with an AI-based text generator and the experiment is repeated. The purpose of this second step is to examine and assess the robustness of the AI-based RFI semantic search algorithm to the noisy generated data. Assessing robustness to noise is necessary in all NLP tasks, as most of the task-specific datasets are not fully representative of the infectivity of the potential user inputs. In the literature, datasets of reference can be perturbed (adding spelling errors, casing, modifying the order of the words or using translation back and forth) either to train the model on noisier data or to evaluate its robustness to noisy inputs [17]. In particular we expose how a limited corpus has been enriched with noisy AI-generated data. The performance and the robustness of the AI model are monitored to be kept similar before and after the data augmentation, while a human-in-the loop qualifies the AI-generated data. A special focus is made on the typos, synonyms and spelling errors in the RFI query.

C. *Implementation architecture of the semantic similarity search on RFIs*

The implementation of the semantic search algorithm consists of 2 steps. The first step is the training phase from historical RFI data (vector space representation of the RFI database), and the second step concerns the inference phase on new RFIs. The training phase consists in building machine learning models that encode the RFI database elements into a vector space representation while preserving the semantics of the corpus. The trained models are then saved and ready to be used in the inference phase. Fig.1 illustrates the 2 phases.

The training phase builds the word embeddings of the corpus associated with the RFI database. This embedding represents an encoding result which is a low-level representation of the RFI corpus into a vector space. Four different algorithms have been implemented for this study. A deep learning-based architecture derived from the encoder transformer’s architecture (Universal Sentence Encoder) [12], a latent semantic indexing algorithm (LSI) [13], a TF-IDF (term-frequency - inverse document frequency) algorithm [14] and finally a fuzzy string match (FSM) algorithm [15].

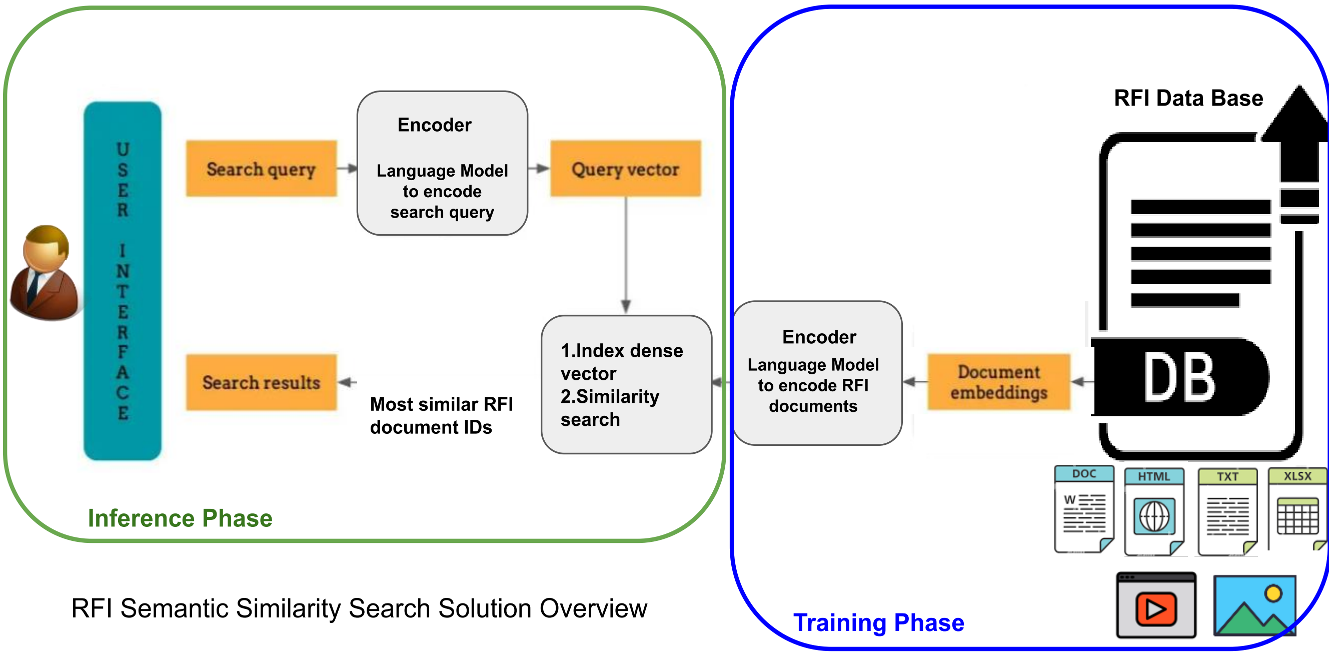


Figure 1: RFI Semantic Similarity Search Solution Overview

For the deep-learning architecture, the context-based encoding method used is the Universal Sentence Encoder, which is a model based on a transformer architecture that considers both the word order and the identification of the remaining word in the sentence. The LSI instead analyses a set of documents in order to discover statistically relevant co-occurrences of words or terms. It uses a document-term matrix decomposition of the corpus of document to cast queries into a low-rank representation vector space, enabling to compute query-document similarity scores in this low-rank representation vector space. The matrix decomposition can be updated with new observations at any time, for an online, incremental, memory-efficient training.

III. Datasets and Experiments Setup

A. Datasets

Handcrafted datasets

A database of 300+ entries of manually created RFIs in the STANAG[2] format have been created for this study. This RFI database has been built by an *intelligence analyst* to better represent the RFI data structure and mandatory fields. The RFI database contains the following fields:

- a synthetic *subject* description or RFI *title*
- *details* if any, corresponding to additional details enriching the RFI *title* (free text).
- areas of interest concerning the needs, describing the *area* or *zone* of interest to focus the search in.
- the authority identifier (*Commander*)
- the recipients in action and in information
- the date parameters “Not Earlier Than” & “Not Later Than”

In the scope of this study, only textual fields have been considered as mandatory: *Subject* and *RFI Details*. The Table I below shows a caption of a realistic RFI sample database.

TABLE I: Example of RFI sample database with FRI *Subject* and *Details* fields. The *intelligence analyst* tries to find semantic similarity between the *commander's* request and the RFI database (*Subject* or *Details*)

Subject	Details
Anti-aircraft threats on Coalition Forces within the operation area XXX	Enemy forces have anti-aircraft means deployed in XXX. It has short- and medium-range anti-aircraft systems dating from the 1980's, including the YYY and ZZZ ground-air defence system. It includes relatively low-tech anti-aircraft guns, to deal with the jets, helicopters, drones, and missiles. What are the threats represented by these assets?
Iraqi theatre - Factors of unrest in the civilian population in the North region	What are the potential factors of unrest in the civilian population in northern Iraq?

[2] https://nso.nato.int/nso/nsdd/main/list-promulg

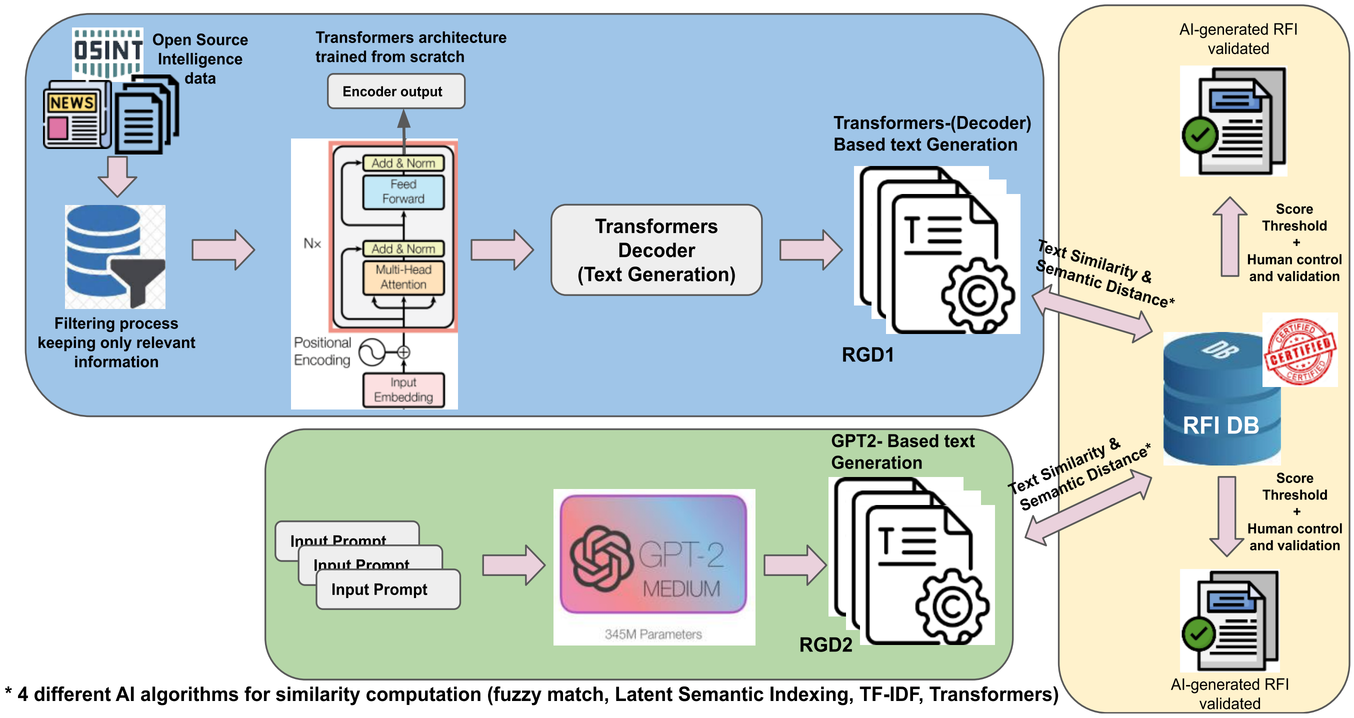

OSINT
Open Source Intelligence data
NEWS
Filtering process keeping only relevant information
Transformers architecture trained from scratch
Encoder output
Add & Norm
Feed Forward
Nx
Add & Norm
Multi-Head Attention
Positional Encoding
Input Embedding
Transformers Decoder (Text Generation)
Transformers-(Decoder) Based text Generation
RGD1
Text Similarity & Semantic Distance*
AI-generated RFI validated
Score Threshold + Human control and validation
DB
CERTIFIED
RFI DB
Score Threshold + Human control and validation
AI-generated RFI validated
Text Similarity & Semantic Distance*
GPT2- Based text Generation
Input Prompt
Input Prompt
Input Prompt
GPT-2 MEDIUM
345M Parameters
RGD2
* 4 different AI algorithms for similarity computation (fuzzy match, Latent Semantic Indexing, TF-IDF, Transformers)

Figure 2: AI-based Intelligence Domain-specific Text Generation and Validation

AI-based text generation for augmenting the reference RFI database

The AI-based intelligence domain specific text generation and validation is described in Figure 2. Two datasets have been generated. The first one (RGD1) has been generated after training from scratch an encoder-decoder architecture (keras_nlp [3] transformer encoder, 12 heads, embedded dimension 64) on open-source intelligence data [4][5][6][7][8][9], including release press papers, domain-specific intelligence journals, intelligence reports, etc. (blue box in Fig. 2). The second dataset (RGD2) has been generated using the GPT2 pre-trained model (green box in Fig. 2). We used the *GPT2-medium*[10] model with 345M parameters. We extracted the *title* of all the RFIs in the *ground truth* database (manually generated RFI datasets) and used them as a prompt for text-generation. For both datasets, the tokens generated have been monitored to be semantically "close" to the RFI *ground truth* database.

For each set of generated tokens, a semantic similarity score has been computed and these tokens are discarded if the score (cosine distance) is less than a predefined threshold in addition to the *human control and validation* (mustard yellow box in Fig. 2). The predefined threshold value will be discussed in the next section. The AI-based text-generated data are then used to augment or enrich the *ground truth* RFI database. From the process described in Fig. 2, we have generated 17,820 additional RFI.

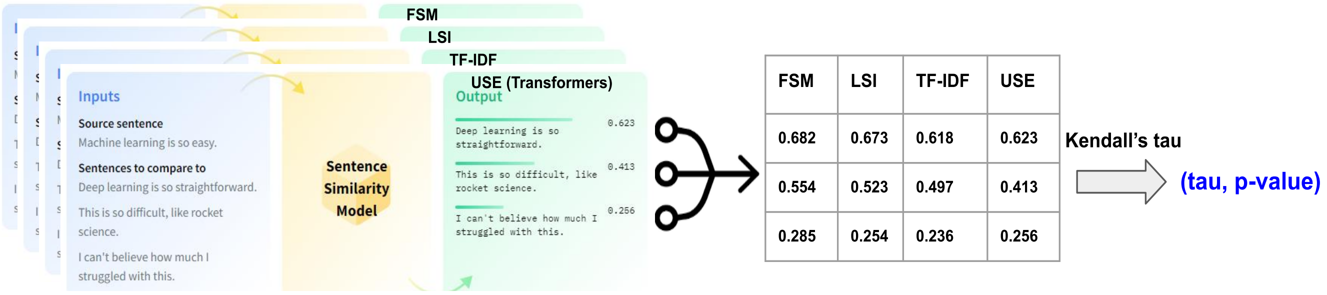
B. *Experiment setup*

In this section, the setup experiments carried out for both the realistic manual generated RFI dataset and the AI-generated RFI data set is discussed. The RFI semantic similarity detection is performed on both datasets and the performances are monitored to be kept similar.

1) *Experiment 1: evaluation on handcrafted data*

The training and inference phases for RFI similarity detection described in section III.D are implemented on the manually generated RFI datasets (ROVEY dataset: RFI Operator Validated EntrY). All of the four algorithms are considered here and a set of semantic search similarity metrics are computed. The common similarity metric used in the semantic search to measure the similarity/distance between vectors have been considered for this study, which is cosine similarity. In addition to this metric, we have also considered the *recall*, measuring the ability of a search engine to find the relevant material in the index, and the *precision*, measuring its ability to place that relevant material high in the ranking. But these latter metrics have only been computed for the first experiment executed on the handcrafted RFI dataset, as the intelligence analyst helped in performing the time-consuming annotation task on this dataset.

[3] https://keras.io/guides/keras_nlp/transformer_pretraining/
[4] https://www.foreignaffairs.com/browse/snapshot
[5] https://intpolicydigest.org/domestic-extremist-groups-pose-a-unique-challenge/
[6] https://intpolicydigest.org/essential-breakthrough-in-kazakhstan-uzbekistan-relations/
[7] https://documents.theblackvault.com/documents/dia/Afghanistan_Stalemate_Continues_CLEAR.pdf
[8] https://nsarchive.gwu.edu/document/29551-42-annual-threat-assessment-switzerland
[9] https://nsarchive.gwu.edu/document/18362-national-security-archive-estonian-foreign
[10] https://huggingface.co/gpt2-medium


FSM
LSI
TF-IDF
USE (Transformers)
Inputs
Source sentence
Machine learning is so easy.
Sentences to compare to
Deep learning is so straightforward.
This is so difficult, like rocket science.
I can't believe how much I struggled with this.
Sentence Similarity Model
Output
0.623
Deep learning is so straightforward.
0.413
This is so difficult, like rocket science.
0.256
I can't believe how much I struggled with this.
Kendall's tau
(tau, p-value)

FSM	LSI	TF-IDF	USE
0.682	0.673	0.618	0.623
0.554	0.523	0.497	0.413
0.285	0.254	0.236	0.256

Figure 3: Robustness evaluation process

2) *Experiment 2: evaluation on synthetic data*

The second experiment implements the same RFI semantic similarity search algorithm on an AI-generated dataset (RFI GPT Data: RGD1&2). The training and inference processes described in experiment 1 are thereafter performed. For Experiment 2, the three similarities metrics described in the previous section below are also computed and they are compared with the ones obtained in Experiment 1.

For each of these two experiments, the robustness of the system is assessed using Kendall's tau correlation coefficient [16] on the similarity search scores for all of the 4 search algorithms (FSM, LSI, TF-IDF, USE). Fig. 3 illustrates the evaluation process. A total of 168 individual queries have been performed, and for each single query and each single algorithm, we retrieve the top-k more similar items in terms of score and we compute the Kendall's tau correlation coefficient between the score tables from one algorithm and another one (e.g., LSI vs TF-IDF). Kendall's tau is a non-parametric measure of relationships between columns of ranked data. The Kendall's tau correlation coefficient returns a value between 0 and 1, where 0 shows no relationship and 1 is a perfect relationship. Kendall's tau has been chosen because it has good statistical properties and its interpretation in terms of the probabilities of observing concordant and discordant pairs is very direct. Thus, if the Kendall's tau is in the same range, before and after the data augmentation, then the semantic search algorithms are robust to data augmentation.

IV. Results

A. *Accuracy*

The accuracy of the search algorithms has been assessed first in terms of the relevance of the RFI responses. For each RFI query, the search results are carefully analyzed by the *intelligence analyst*. They then attribute a binary rating (0: Bad response, 1: Good response) to the overall search results in a very conservative way, and provides some free-text comments about the accuracy, the relevance and the robustness of these results. Each query is considered as one trial. The similarity score ranges from 0 (no similarity) to 1 (perfect match). 45 individual trials (query) have been performed and a naive statistical count of the good responses versus the bad ones have been done. We obtained 36 good responses and 9 bad responses which gives a global "accuracy" score of 80% with a database of 300 RFI entries (ROVEY dataset).

B. *Robustness*

A set of 168 individual queries have been performed for each of the four similarity algorithms; only the top-20 similar RFIs have been selected for each query and the threshold value of the similarity score (cosine similarity) has been fixed to 0.35. If a result has a score inferior to this threshold value, we do not consider it (i.e., no relevant match). In addition, we only consider query results with more than five RFI responses to have "enough" sample data to compute Kendall's tau coefficient. In order for a response to be considered, it has to be captured by each of the four algorithms. Table II represents an illustration of the ranked similarity score obtained for one query for each of the four algorithms. In this example, the query's response showed 11 similar RFIs with similarity scores greater than 0.35. The Kendall's tau coefficient and the associated p-values are computed considering pairs of data in Table II. The results are shown in Table III.

TABLE II: Ranked similarity scores for all of the four algorithms (single query)

	Fuzzy	LSI	TF-IDF	USE
0	1	1	0.89	1
1	0.92	0.92	0.82	0.95
2	0.86	0.92	0.82	0.92
3	0.81	0.92	0.73	0.87
4	0.79	0.92	0.73	0.85
5	0.77	0.85	0.73	0.82
6	0.74	0.84	0.56	0.77
7	0.63	0.84	0.49	0.71
8	0.61	0.84	0.49	0.64
9	0.61	0.84	0.46	0.58
10	0.6	0.78	0.43	0.57

TABLE III: Kendall's Tau coefficient and the associated p-values for score data in Table II (to read (Tau, p-value))

	Fuzzy	LSI	TF-IDF	USE
Fuzzy	(1.0, 2e-05)	(0.892, 0.00039)	(0.943, 0.0001)	(0.991, 2e-05)
LSI		(1.0, 0.00019)	(0.884, 0.00061)	(0.884, 0.0004)
TF-IDF			(1.0, 6e-05)	(0.953, 7e-05)
USE				(1.0, 0.0)

Table III shows on average a quite high Kendall's tau correlation coefficient (with minimum value greater than 0.88), meaning a strong correlation between all of these algorithms. Moreover, the p-values are low (close to 0), thus rejecting the null hypothesis (that there would be an absence of association between two algorithms). Kendall's non-parametric statistical test has been considered to succeed if Kendall's tau value is greater than 0.8 and the p-value less than 0.03.

For the original RFI database (ROVEY dataset), among the 168 individual queries performed for each algorithm, 84 retrieved at least 5 similar RFI with a similarity score greater than 0.35. Moreover, these figures were the same for all of the four algorithms, leading to a 50% "good" RFI query matching. Among these 84 queries, all of them passed the Kendall test, which is Kendall's tau coefficient is greater than 0.8 and its p-value less than 0.03 as stated above. The same experiment has been performed after the RFI data generation and augmentation. From the process described in Fig. 2, we have generated 17820 additional RFI (RGD data), considered as noise in the original ROVEY dataset, as the upper bound of the similarity score with the later DB was kept quite low at 0.2 in order to get more RFI entries to enrich the original DB. Summary of these results are presented in Table IV, where we exhibit results with and without data augmentation. As the RFI generated data are considered as noise in the original RFI database, the figures in Table III show that the different performance indicators do not evolve at all before and after the data augmentation, which makes all of the similarity search models robust to noisy data augmentation. These stable results are first due to the noisy nature of the generated data, and secondly, due to the upper bound value of the similarity score for RFI data generation (0.2) which is less than the threshold score value considered for all of the query's responses (0.35).

TABLE IV: Overall Key Performance Indicators (KPI) with and without data augmentation

Parameters to be monitored	**RFI DB without Data Augmentation**	**RFI DB with Data Augmentation**
Total Nb or queries	168	168
Nb of queries that crossed the score threshold[11]	84	85
Nb of queries that passed the Kendall's statistical test	84	85
Nb of queries that failed the Kendall's statistical test	0	0

Another figures to look at are how the system behaves with and without the data augmentation. To do so, among all the queries that passed the score threshold and Kendall's statistical test, we identified the intersection between the 84 and 85 responses queries. This intersection was identified based on the similarity in the query's input text for the search and the equivalence in the output results. From this subset of similar queries responses between RFI DB with and without data augmentation, we computed for each individual query, how the number of retrieved RFI document behaved. In other words, we check whether the number of retrieved RFIs changes before and after data augmentation, taking as reference this number before data augmentation. We also focus on whether the order (ranked by similarity score) of the responses is preserved before and after the data augmentation. Table V shows that a high percentage of RFI responses were observed both before and after the data augmentation (85.7%), among them 66.7% preserved the number of RFI retrieved, while 30.5% led to the same number of RFI responses, before and after the data augmentation. The order in terms of ranking of the similarity score was preserved in 23.6% of cases.

TABLE V: KPI at individual query's response level, taking as reference the RFI database without data augmentation

KPI - Intersection	**KPI - Inclusion**	**KPI - Preserve same Nb of RFI responses**	**KPI - Preserve same order in RFI responses**
72	48	22	17
85.7%	66.7%	30.5%	23.6%

KPI - Intersection: among the (84, 85) pairs of queries that passed all the tests (threshold score and Kendall's test), how many individual response pairs passed the test?

KPI - Inclusion: among the Intersection, how many preserved the number of RFI responses compared to the results without data augmentation?

KPI - Preserve same Nb of RFI responses: among Intersection, how many preserved the same number of RFI responses compared to the results without data augmentation?

KPI - Preserve same order in RFI responses: among the Intersection, how many preserved the same order (ranked similarity score) in RFI responses compared to the results without data augmentation?

V. Discussions

The two experiments performed in this study showed quite similar results both qualitatively and quantitatively for the RFI semantic search task. The overall model accuracy was estimated at 80% for the semantic search task on ROVEY dataset, while the robustness to noise was assessed using various techniques (Kendall's tau and statistical analysis of queries output). The semantic similarity search score threshold was fixed at 0.35 to consider only relevant RFI responses to compute the performance metrics, and the RFI response's selection process was rather conservative both in terms of Kendall's tau value and its associated p-value. The metrics displayed in Tables III, IV and V show an overall good performance of our approach, making it a good candidate to tackle the lack of data to train AI models on in the defence sector.

[11] Similarity scores greater than 0.35 for each query's response and each algorithm and a minimum of 5 similar RFI retrieved for each algorithm

VI. Conclusion and future work

The study examined how an intelligence domain specific AI-based text-generator can be used to augment or enrich a *ground truth* database which supports the RFI semantic similarity search. In particular we exposed how a limited RFI corpus has been enriched with noisy AI-generated data, and assessed the performance and the robustness of the AI model to this noisy data when performing RFI semantic similarity search. We assessed the performances of the search engine and showed that they were kept similar before and after the data augmentation, making the search algorithms robust to data augmentation in particular noisy data. The assessment relied on a *human-in-the loop* (*intelligence analyst*) who annotated and qualified both the handcrafted data and the AI-generated data. The different results discussed in this study make the proposed approach a good candidate to deal with the lack of data when designing and training AI systems in the defence sector. It is worth mentioning that the approach was tested on handcrafted realistic RFI data, and needs to be validated on real data from operations. Moreover, the annotation step and model accuracy assessment would have to be performed by different *end-users* in order to reduce as much as possible the underlying bias. These open questions have to be addressed in future studies.

Acknowledgment

This project has received funding from the European Defence Industrial Development Programme (EDIDP) under grant agreement No: EDIDP-AI-2020-066-AI4DEF.

References

[1] Sébastien Bubeck, «Sparks of artificial general intelligence: Early experiments with gpt-4» *arXiv* , 2023.

[2] Aakanksha Chowdhery, «Palm: Scaling language modelling with Pathways» *arXiv*, 2022.

[3] Hugo Touvron et al, «LLaMA: Open and Efficient Foundation Language Models» *arXiv*, 2023.

[4] OpenAI, «Introducing chatgpt,» 2022. [En ligne]. Available: https://openai.com/blog/chatgpt.

[5] OpenAI, «GPT-4 Technical Report» *arXiv* , 2023.

[6] A. Radford et al. «Improving Language Understanding by Generative Pre-Training» 2018.

[7] A. Radford et al., «Language Models are Unsupervised Multitask Learners,» 2018.

[8] T.B. Brown et al., « Language models are few-shot learners» *arXiv* , 2020.

[9] A. Vaswani et al. «Attention is all you need» *Advances in Neural information processing systems,* vol. 30, n°130, 2017.

[10] Haifeng Wang et al. « Pre-Trained Language Models and Their Applications» *Engineering,* n° %1 ISSN 2095-8099, 2022.

[11] J. Hochreiter et al. «Long short-term memory,» *Neural computation*, vol. 8, n°19, pp. 1735-1780, 1997.

[12] Daniel Cer et al. , «Universal Sentence Encoder,» *arXiv,* 2018.

[13] S. Deerwester et al., «Indexing by latent semantic analysis» *Journal of the American Society for Information Science,* n°141, 1990.

[14] G. Salton et al., «Term-weighting approaches in automatic text retrieval» *In Information Processing & Management,* vol. 24, n°15, pp. 513-523, 1988.

[15] Rares Vernica et al., «Efficient top-k algorithms for fuzzy search in string collections» *Proceedings of the First International Workshop on Keyword Search on Structured Data,* pp. 9-14, June 2009.

[16] L. Laurencelle, « Le tau et le tau-b de Kendall pour la corrélation» *Tutorials in Quantitative Methods for Psychology,* vol. 5, n°12, pp. 51-58, 2009.

[17] J. Náplava, et al. «Understanding model robustness to user-generated noisy texts» *arXiv* , 2021.