Surface & Coatings Technology 466 (2023) 129680

Contents lists available at ScienceDirect

Surface & Coatings Technology

journal homepage: www.elsevier.com/locate/surfcoat

Generative pre-trained transformers (GPT) for surface engineering

Spyros Kamnis

Castolin Eutectic-Monitor Coatings Ltd, Newcastle NE29 8SE, UK

ARTICLE INFO

Keywords:
Thermal spray
GPT
Generative AI
LLM
Large language model
NLP

ABSTRACT

The knowledge of scientific articles within Generative Pre-trained Transformers (GPT) is not exhaustive due to factors such as data coverage, freshness, complexity, paywalls, and context. While it can provide general information on scientific topics, it may struggle with specialized terminology, recent research, and nuanced understanding. As a result, relying on GPT as a scientific assistant tool may not be ideal. Instead, it is important to consult specialized resources and databases for a comprehensive understanding of specific scientific domains and access to the latest research. A custom data driven GPT can enhance its performance as a scientific assistant tool by improving domain knowledge, providing up-to-date information, reducing ambiguity and errors, performing customized tasks, and offering enhanced search capabilities. This work demonstrates and evaluates the use of such GPT models using a small selection of peer reviewed published thermal spray articles as the reference domain knowledge. The specific domain knowledge model works exceptionally well outperforming the general state-of-the-art large language models.

1. Introduction

The emergence of advanced natural language processing techniques has revolutionized the way artificial intelligence is interacted with and utilized. Among these innovations, Generative Pre-trained Transformers (GPT) [1–3] have emerged as a game-changer, offering unprecedented capabilities in generating human-like text and understanding natural language. However, the effectiveness of general-purpose GPT models in specialized domains may be limited due to a lack of domain-specific knowledge and contextual understanding.

GPT models, or Generative Pre-trained Transformers, are trained using a two-step process that involves pre-training and fine-tuning. This approach is common in large language models, as it enables them to learn general language understanding and then specialize in specific tasks or domains. During the pre-training phase, GPT models are trained on vast amounts of text data collected from diverse sources like websites, books, and articles. The model learns to predict the next word in a sentence given the previous words, a process known as masked language modeling. In this phase, the model captures grammar, syntax, semantics, and acquires a significant amount of general knowledge. The pre-training phase is computationally intensive and requires extensive resources, but it only needs to be done once for each model architecture.

After pre-training, GPT models are adapted to specific tasks or domains by fine-tuning them on smaller, task-specific datasets. This process involves training the model for a few additional epochs using labeled data from the target domain or task, such as sentiment analysis, text summarization, or question-answering. Fine-tuning allows the model to specialize in the desired task and improve its performance on domain-specific data while retaining the general language understanding learned during pre-training. The architecture of GPT models is based on the Transformer, which was introduced by Vaswani et al. in 2017 [4]. Transformers employ a self-attention mechanism that allows them to process and learn long-range dependencies in text more effectively than previous models like RNNs [5] and LSTMs [6]. This architecture has become the foundation for many state-of-the-art Neuro-linguistic programming (NLP) models as it is highly scalable and parallelizable, enabling the training of increasingly large models.

This article dives deeper into the concept of custom data indexing GPT models and not fine tune training on domain-specific data, enabling organizations, researchers, and professionals to tap into the full potential of these powerful AI tools in their respective fields. Data indexing refers to the process of organizing and storing data in a way that makes it easier and faster to search and retrieve relevant information. In the context of large language models (LLMs), like GPT-4 from OpenAI, data indexing can offer several advantages over training full LLMs, though it is important to note that these two concepts serve different purposes and have distinct strengths and weaknesses.

Fine tune training a full LLM in a specific domain requires a significant amount of time, computational resources, and energy. Data indexing, on the other hand, is typically less resource-intensive and can

E-mail address: s_kamnis@hotmail.com.

https://doi.org/10.1016/j.surfcoat.2023.129680
Received 23 April 2023; Received in revised form 29 May 2023; Accepted 30 May 2023
Available online 2 June 2023

be completed more quickly. This makes data indexing more cost-effective and efficient for certain tasks. Data indexing allows also for more compact storage, as it only stores relevant information and the relationships between the data points.

Scalability and customization are the main two benefits of data indexing instead of model training especially when dealing with ever-growing datasets. Indexing systems can be easily updated with new information, while retraining an entire LLM can be a lengthy and resource-intensive process. Finally, indexing systems can be tailored to specific domains or applications, allowing for more accurate and relevant results. Training a full LLM, on the other hand, requires learning from a wide range of data sources, which can result in a more generalized and less specialized understanding of the subject matter. Data indexing allows for faster and more targeted information retrieval through the use of specific queries. This enables users to find precise information more easily, whereas an LLM might generate more generic responses that require additional interaction or clarification.

Thermal spray processes involve a multitude of parameters, such as spray distance, powder feed rate, gas pressure, gas flow rate, and temperature, all of which contribute to the final coating characteristics. Managing these parameters effectively is often complex and necessitates a fine balance. In practice the data-indexed GPT models can ingest and learn from vast amounts of historical and real-time process data, creating a comprehensive understanding of the correlations between these parameters and the quality of the coatings. With this understanding, the model can provide optimal parameters leading to reducing scrap rates, improving efficiency, and enhancing the overall coating quality. Moreover, data-indexed GPT models offer predictive capabilities, allowing operators to anticipate issues before they arise. For instance, the model might predict the risk of coating defects or equipment failure based on certain parameter combinations or process conditions, enabling preventive actions.

Furthermore, through continuous learning, these models can offer incremental improvements to the process over time. They can identify patterns and trends that might not be apparent to human observers and suggest process modifications that further optimize performance and output quality. Thus, the application of data-indexed GPT models to thermal spray processes can substantially improve operational efficiency, product quality, and preventive maintenance. This embodies a step forward towards Industry 4.0, where AI plays an integral role in process optimization and decision-making.

In addition to process optimization in the context of Industry 4.0, data indexing helps organize and classify data in a structured manner, which in turn assists in better comprehension of the data and associated technology. With indexed data, users can easily access relevant information, see relationships between data points, and derive insights more efficiently. In the context of technology, this can make complex systems more approachable and understandable, aiding in problem-solving, decision-making, and innovation.

Significant benefits are also expected in training and education. Indexed models like GPT can be used to create interactive learning experiences. These models can generate human-like text based on the input they receive, making them useful for creating training materials, answering questions, and providing explanations. For instance, an indexed GPT model could ingest a large dataset of technical manuals and then be used to answer queries about the content of those manuals, effectively assisting in training. It's also feasible that such models could be used to create dynamic, personalized learning experiences that adapt to a learner's progress and needs.

In this article the process of creating a custom GPT model for thermal spray processes is explored, highlighting the steps involved and the challenges and limitations one might encounter during this process along with potential solutions.

2. Method

Large Language Models (LLM) like GPT [1], are pre-trained on vast amounts of publicly available data. To incorporate private data, there are two main paradigms: in-context learning and fine-tuning. In this article we focus on In-context learning that involves inserting context into the input prompt, allowing the LLM to generate a response by leveraging its reasoning capabilities. In terms of data connectors, LlamaIndex provides connectors to various data sources and formats, such as APIs, PDFs, documents, and SQL databases. The platform creates indices for both unstructured and structured data to facilitate in-context learning. These indices help with storing context for easy access and prompt insertion, handling prompt limitations when the context is too large and managing text splitting.

2.1. Indexing

To efficiently enhance Large Language Models (LLMs) with private data indexing, many options are available. Indices play a crucial role in organizing and retrieving data efficiently, especially when working with large datasets or complex structures. In this context, there are three main types of indices (Fig. 1), each serving a different purpose in data organization and retrieval.

List Index (Fig. 1a) is the simplest form of indexing, it stores nodes sequentially in a list. In the realm of NLP and machine learning, 'nodes' represent individual data points, which could be sentences, paragraphs, or documents in a text dataset. This type of index is suitable for small datasets where the retrieval of data can be performed linearly. However, for larger datasets, searching through a list index can be time-consuming and inefficient. The Vector store index (Fig. 1.b) is designed to store nodes along with their corresponding embeddings in a vector space. Embeddings are compact representations that capture the semantic meaning of the text. Storing nodes in a vector space enables efficient similarity-based retrieval of data. During querying, the most similar nodes to the query are identified and passed to the response synthesis module. The response synthesis is a process where the model generates a response or output based on the input it receives and the data it has been trained on. This type of index is particularly useful when working with large-scale datasets where semantic similarity between the query and nodes is important.

In the context of scientific queries, vector store index can efficiently retrieve articles with similar content and context. This is especially helpful when users are looking for articles related to a specific topic or research area. In addition, the Tree Index method (Fig. 1.c) can be used if the documents have a hierarchical structure, such as categories, subcategories, or topics. This can improve search efficiency and help users explore related articles in a structured manner.

The response synthesis in linear and vector indices involves the use of the context in the first node, along with the query, to generate an initial answer. Then pass this answer, the query, and the context of the second node as input into a "refine prompt" to generate a refined answer. The refine process continues through N-1 nodes, where N is the total number of nodes. In this way the external information is organized in a way that is compatible with GPT LLM prompt limitations. This work uses Langchain's LLM and LLMChain modules and implements the LLMPredictor wrapper class for integration. Although OpenAI's text-davinci-003-GPT3.5 model [7] is used in this work, the LlamaIndex toolkit [8] supports LLM customization, such as changing the underlying LLM, modifying output tokens, and fine-tuning other parameters. The indexing scripts are provided in Appendix A.

2.2. Answer quality evaluation method

Recently LlamaIndex [8] introduced an indexing and evaluation framework designed for language models like GPT. The evaluation components within LlamaIndex do not require ground-truth labels for

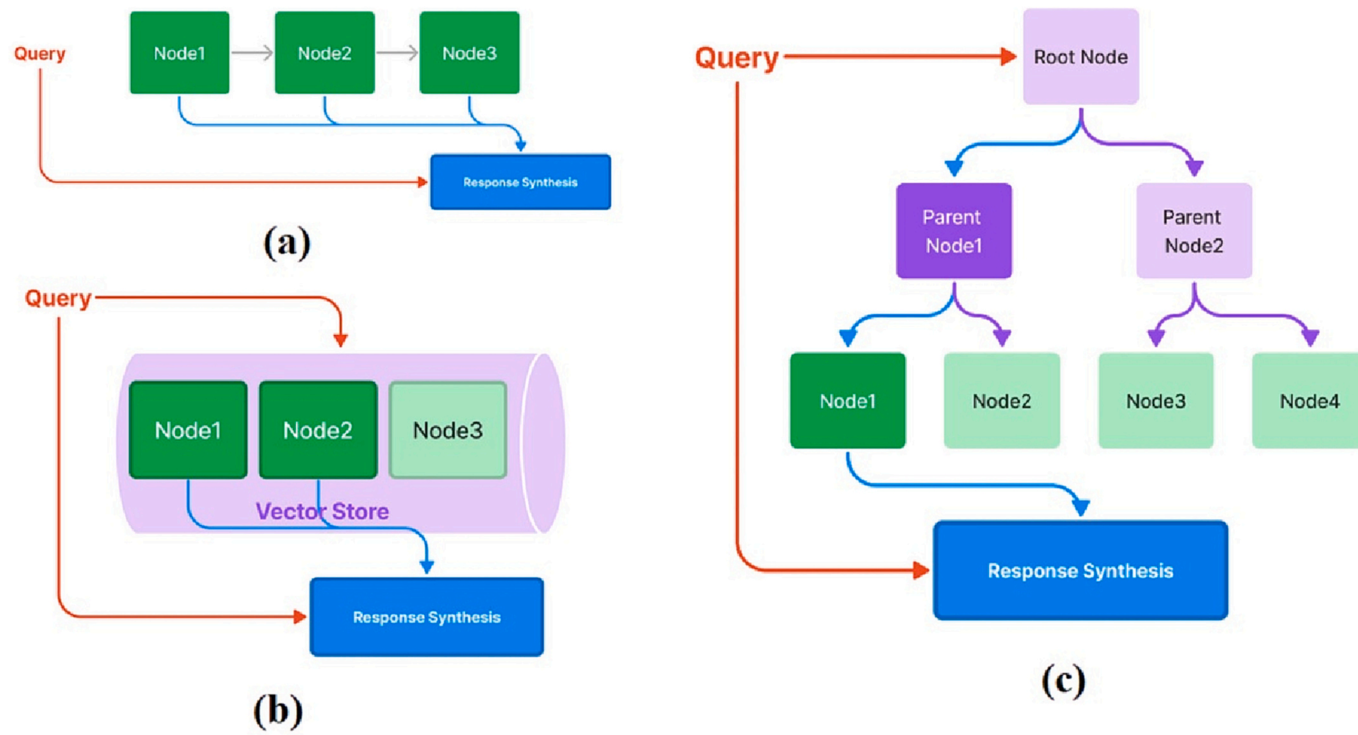

Fig. 1. The node corresponds to a chunk of text from a document which is parsed into node objects. Response Synthesis is the process where the module which synthesizes a response provides the answer to the user a) Linear Index, b) Vector Index, c) Tree Index. [8]

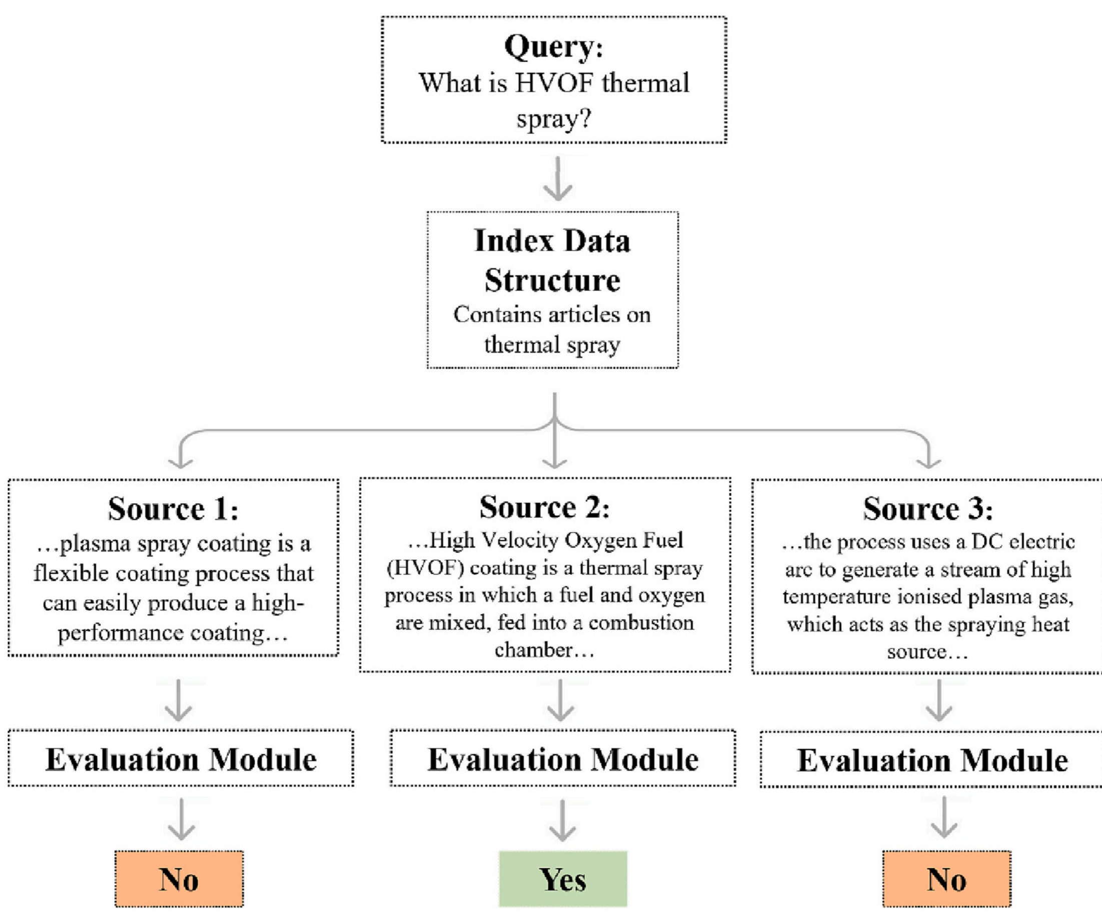

Fig. 2. Response evaluation framework. When the model returns no means that the answer is not relevant to query due to missing information in the source node which contains the indexed data. When the answer is 'yes' the model confidence is high regarding the given answer to a query.

assessment, making it versatile and adaptable to various use cases. The evaluation process can utilize the query, context, and response, as well as make additional calls to the language model as needed.

The evaluation process begins with a query, which is a question or a request for information. The query is a key component of the evaluation, as it defines the problem that the language model is trying to solve. Context, on the other hand, is the background information or additional details that may be relevant to the query. The evaluation components can utilize the context to assess the language model's understanding of the problem and its ability to provide a meaningful response.

The response is the output generated by the language model in response to the query. The evaluation components can analyze the response to determine its quality, relevance, coherence, and other attributes. During the evaluation process, additional calls are made to the language model to gather more information, generate alternative responses, or assess different aspects of the model's performance as shown in Fig. 2. These calls can be used to further refine the evaluation and provide a more comprehensive assessment of the language model. By leveraging the query, context, response, and additional language model calls, a flexible and powerful evaluation framework is provided herein that is adaptable to a wide range of use cases and scenarios (Appendix A).

2.3. Thermal spray data

Data ingestion and transformation are two important steps in the data processing pipeline. Data ingestion refers to the process of collecting, importing, and transferring raw data from disparate sources into a central repository. This can involve data from various formats, such as text files, spreadsheets, databases, and more. The goal is to make the data accessible and ready for further processing, analysis, or storage.

Data transformation, on the other hand, is the process of converting raw data into a structured format suitable for analysis or machine learning models. This typically involves cleaning, formatting, and organizing the data, as well as potentially enriching it with additional information. In the context of language models like LLM, data transformation is crucial for ensuring that the input data is in a suitable format for fine-tuning the model.

In this article, publications [9–54] with diverse thermal spray content in a PDF format, were used without any pre-processing. The dataset was constructed using familiar publications to ensure accurate evaluation of the model's responses, maintain consistency in style and terminology, and keep the content relevant and up-to-date. The PDF documents with structured language were converted into a format that LLM can understand by using the python parsing library PyPDF2 [55]. Since pre-processing is not practical for large datasets, in this work the PDF were not cleaned by removing irrelevant characters, such as special symbols, numbers, or punctuation marks.

In the scope of this study, the original text was tokenized, breaking it into individual words or tokens, using the standard NLTK tokenization technique [56]. Tokenization is a vital step in NLP, as it helps the model understand and manipulate the text at a more granular level. Once the text was broken into tokens, each token was assigned a unique identifier, or token ID. The resulting dataset consisted of pairs of tokens and their corresponding token IDs. In these pairs, the tokens themselves served as the 'keys', and the token IDs served as the 'values'. This process of assigning IDs to tokens is an essential part of preparing text data for analysis by an LLM, as it translates the text into a format that the model can work with.

Following this, the data were converted into a sequence of integers corresponding to these token IDs and saved in the JSON format [57]. The JSON format is a widely used data interchange format that represents data as key-value pairs. In our context, the 'keys' were the tokens from the original text, and the 'values' were their corresponding token IDs. Once the PDF documents were converted into this token-to-token-ID format, they were fed into the Large Language Model (LLM). This allowed the LLM to analyze, generate, or classify text based on the input data, essentially enabling the model to 'understand' and 'work with' the text from the PDF documents."

3. Results & future work

3.1. Model comparisons

In this section, the performance of GPT-4 and the fine-tuned, data-indexed GPT-3 model in answering a variety of queries is evaluated. The results are presented in Table 1, which uses the vector indexing method, and Table 2, which employs the tree-indexing method. The human expert and self-evaluation responses are provided, along with the source text utilized to generate the answer. The source text content is automatically extracted from the dataset by the model. A beta version of the model is available as a web app and can be accessed at the following link: https://hvof.pythonanywhere.com/.

Query 1 relates to the addition of Mo in the Cantor High Entropy Alloy. The disparity in responses from the original GPT-4 model and the fine-tuned, data-indexed model underscores the benefits of fine-tuning and domain adaptation. The response from the fine-tuned model is more precise and comprehensive, specifically mentioning the formation of new phases, such as BCC solid solution, γ-phase, and Laves, and referencing the XRD analysis found in the indexed content. Furthermore, the self-evaluation method accurately determines that the answer is relevant to the input query.

In contrast, the original GPT-4 model offers a less accurate response, incorrectly stating that the promotion of the bcc structure alone is responsible for the enhanced properties. The heightened accuracy in the fine-tuned model's response can be directly attributed to its domain-specific knowledge and superior access to pertinent information. When the tree indexing method (Table 2) is utilized, the model is unable to locate the relevant source and returns no answer. It is evident that vector indexing demonstrates an improvement over the tree method in this instance.

In Query 2, the model's reasoning ability to formulate an answer to a general knowledge question is evaluated. The GPT-4 model, with a knowledge cut-off after 2021, was unable to answer this question. However, the vector indexing method accurately identified the relevant source in the dataset and provided the correct answer. The self-evaluation function operated as expected, recognizing the answer as relevant and correct. Conversely, the tree indexing method returned an irrelevant answer. The identified source was unrelated to the query; nevertheless, the self-evaluation algorithm acknowledged that the query answer and context were not relevant and returned 'NO,' indicating that this answer should not be trusted.

Query 3 asks for a hardness value for a specific high entropy alloy composition provided in the indexed dataset. Both the GPT-4 model and the tree-indexed method arrived at correct conclusions but did not supply a specific hardness value. The vector indexing method precisely identified the relevant source text and furnished a comprehensive answer. In all instances, the self-evaluation algorithm accurately assessed the answer's relevance to the query and context information.

3.2. Prospects of fine-tuning foundation models in specific domain knowledge

Foundation models such as GPT and BERT, are pre-trained on extensive datasets, learning to generate text by capturing the underlying structure and semantics of human language. The process of fine-tuning these models involves training them on domain-specific data to adapt their performance for particular tasks or applications. Notably, fine-tuning foundation models offers a variety of benefits. For instance, these models can comprehend context, generating coherent, relevant, and contextually appropriate responses. This capability allows them to provide more accurate and meaningful results compared to conventional

Table 1
Vector Index method and comparisons with OpenAI GPT-4 model. Next to the model response, human evaluation is provided as YES for the correct answer and NO WHEN the answer is not correct or incomplete.

Item	Query	GPT-4		Vector Index GPT		Self-evaluation	Source text
1	What impact does the incorporation of molybdenum (Mo) have on the microstructure and phase formation of Cantor high-entropy alloy (HEA)? Answer in 40 words.	Incorporating Mo into Cantor HEA refines the microstructure, promoting the formation of a dual-phase matrix with face-centered cubic (FCC) and body-centered cubic (BCC) structures. This leads to enhanced mechanical properties, such as increased hardness and improved high-temperature stability.	NO	The addition of Mo to Cantor HEA modifies the single-phase FCC solid solution structure of the original alloy, leading to the formation of new phases such as BCC solid solution, γ-phase, and Laves, as well as the possible existence of intermetallic phases, as observed in the XRD analysis.	YES	YES	Citation: Karantzalis, A.E.; Poulia, A.; Kamnis, S.; Sfikas, A.; Fotsis, A.; Georgatis, E. Modification of Cantor High Entropy Alloy by the Addition of Mo and Nb: Microstructure Evaluation, Nanoindentation-Based Mechanical Properties, and Sliding Wear Response Assessment. Alloys 2022, 1, 70–92. https://doi.org/10.3390/alloys1010006 …
2	Are there any general-purpose thermal spray coating deposition software simulators?	As of my last knowledge update in September 2021, there were no specific general-purpose thermal spray coating deposition software simulators available in the market.	NO	Yes, there is a general-purpose thermal spray coating deposition software simulator that supports either a cylindrical (parallel) beam or a conical beam, under any kinematic model (rotating object and/or gun motion).	YES	YES	Contents lists available at ScienceDirect Surface & Coatings Technology journalhomepage: www.elsevier.com/locate/surfcoat A general-purpose spray coating deposition software simulator Maria Tzinavaa, Konstantinos Delibasisa, Bryan Allcockb, Spyros Kamnisc
3	What is the hardness of CoCrFeMnNi0.8 V alloy produced by vacuum arc melting, and why is its hardness greater than that of the CoCrFeMnNi alloy?	The hardness of the CoCrFeMnNi0.8 V alloy produced by vacuum arc melting can vary depending on the specific processing conditions, cooling rates, and possible heat treatments. However, the addition of vanadium (V) to the base CoCrFeMnNi alloy can result in increased hardness compared to the original alloy.	YES	The hardness of CoCrFeMnNi0.8 V by vacuum arc melting is 13.09 GPa, which is nearly 4 times higher than that of CoCrFeMnNi. This is due to the presence of V-rich particles in the microstructure of CoCrFeMnNi0.8 V, which are finely and homogeneously distributed across the sample and not present in CoCrFeMnNi. These particles increase the hardness of the alloy.	YES	YES	Materials Letters 331 (2023) 133504 Available online 14 November 2022 0167-577X/© 2022 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).Wear resistant CoCrFeMnNi 0.8 V high entropy alloy with multi length-scale hierarchical microstructure

Table 2
Tree index method.

Item	Query	Tree index GPT		Self-evaluation	Source text
1	What impact does the incorporation of molybdenum (Mo) have on the microstructure and phase formation of Cantor high-entropy alloy (HEA)? Answer in 40 words.	ANSWER: None of the above. None of the summaries provided discuss the impact of molybdenum (Mo) on the microstructure and phase formation of Cantor high-entropy alloy (HEA).	NO	None	No source available
2	Are there any general-purpose thermal spray coating deposition software simulators?	Yes, there are general-purpose thermal spray coating deposition software simulators. These simulators can be used to predict the in-flight particle behavior in a liquid fuelled HVOF thermal spray gun, as well as the locations of particle deposition. They can also be used to examine the effect of particle morphology on particle dynamics, and to control coating thickness and deposition efficiency.	NO	NO	shape 60 μm particles. Fig. 8. Temperature evolutions for different size particles. Fig. 9. Particle temperatures at impact with a standoff distance of 32.5 cm.3489 S. Gu, S. Kamnis/ Surface & Coatings Technology 203 (2009) 3485–3490 surface temperature increases slowly as the solidus isothermal surface moves towards the center of the particle. The particle never reaches the fully liquid state since its surface temperature does not rise quickly enough to reach the limiting value. Prior to impa…
3	What is the hardness of CoCrFeMnNi0.8 V alloy produced by vacuum arc melting, and why is its hardness greater than that of the CoCrFeMnNi alloy?	The hardness of CoCrFeMnNi0.8 V by vacuum arc melting is higher than that of CoCrFeMnNi because the addition of 0.8 % V increases the strength of the alloy due to its high melting point and its ability to form hard carbides. This increases the hardness of the alloy and makes it more resistant to wear and tear	YES	YES	The numerical model results during melting and solidification are compared with the developed analytical solution to identify the accuracy of the correction function. The analytical solution can be implemented when the material change's phase at a single temperature…

keyword-based data indexing methods. Moreover, foundation models can leverage transfer learning, applying knowledge acquired during pre-training to numerous downstream tasks. This versatility enables them to manage diverse applications, even when domain-specific data is limited. Furthermore, foundation models excel at handling ambiguity in queries by understanding the underlying semantics and generating suitable responses based on context.

On the other hand, data indexing constitutes a traditional approach to information retrieval, wherein documents or data are organized and indexed according to specific keywords or attributes. This organization facilitates efficient querying and retrieval of pertinent information. Data indexing presents several advantages, such as scalability; compared to fine-tuning foundation models, data indexing requires fewer computational resources and can more efficiently manage large datasets. Additionally, data indexing allows for enhanced control over precision and recall in information retrieval tasks, as it depends on explicit keyword matching and can be adjusted to prioritize relevant documents. Furthermore, data indexing is more transparent and interpretable than foundation models, since it relies on well-defined keyword matching and ranking algorithms, making it easier to comprehend and troubleshoot.

Although both approaches exhibit unique merits, they also possess key distinctions. For example, the complexity of fine-tuning foundation models demands significant computational resources and deep learning expertise, rendering it less accessible for smaller organizations or projects. On the other hand, data indexing is relatively simpler and more direct to implement. In terms of versatility, foundation models can accommodate a broad range of tasks, while data indexing is primarily tailored for information retrieval tasks. When it comes to robustness, foundation models typically perform better in processing ambiguous or complex queries, whereas data indexing may struggle due to its reliance on keyword matching.

The choice between these two approaches hinges on a project's specific requirements, available resources, and the desired level of complexity and interpretability. As research in natural language processing progresses, it will be crucial to harness the strengths of both methodologies to develop hybrid solutions that amalgamate the best aspects of each approach.

3.3. *Prospects of data pipeline for creating a complete model*

In order to construct a custom GPT model with specialized domain knowledge in thermal spray technology, a comprehensive and up-to-date collection of documents pertaining to the subject is essential. Various types of documents can be incorporated in this endeavor, ensuring a diverse and rich training dataset for the model. One important source of information is research articles published in reputable journals related to thermal spray technology, materials science, and surface engineering. Open access peer-reviewed articles offer valuable insights into the latest findings and advancements in the field. Academic databases or specialized databases like ASM International's can be employed to search for relevant articles.

Conference proceedings are another valuable resource, as they contain papers presented at conferences and symposiums that focus on thermal spray, coatings, and surface engineering. These proceedings often encompass the most recent research and developments in the field, making them a vital addition to the training dataset.

Technical reports available in the internet from organizations, research institutions, or industries involved in the development and application of thermal spray technologies can offer practical knowledge and applications of the technology. Similarly, patents related to thermal spray processes, equipment, and materials can provide insights into the latest innovations and advancements. Books and book chapters authored by experts in thermal spray and related fields can contribute to a solid foundation and comprehensive understanding of the subject. Including these sources, respecting always any licensing agreements, can help to ensure the GPT model acquires a deep understanding of the domain.

Industry standards and guidelines, such as those from ISO, ASTM, or ASM International, can be included to address thermal spray processes, materials, and quality control. This information is crucial to ensure the model is well-versed in the best practices and compliance requirements of the industry. Reputable websites, blogs, and forums dedicated to thermal spray technology can also be valuable sources of information. These platforms often feature professionals, researchers, and enthusiasts sharing their knowledge and experiences, offering practical and real-world insights. Lastly, open access educational materials like lecture notes, presentations, or course materials from academic institutions or professional organizations offering courses or training on thermal spray technology can further enrich the training dataset.

Upon gathering a substantial amount of domain-specific data, it is crucial to preprocess the data by cleaning, formatting, and tokenizing the text. With the preprocessed data, the fine-tuning process of the generative LLM can commence, ultimately enhancing its knowledge and understanding of thermal spray technology.

3.4. *The need for answer relevance evaluation*

Answer evaluation is a crucial aspect of assessing the performance of Large Language Models (LLMs) in generating relevant, accurate, and contextually appropriate responses to user queries. This task poses several challenges due to the inherent complexity and limitations of LLMs.

Ambiguity in queries poses a challenge for LLMs, as they often struggle to interpret and generate relevant answers. Additionally, LLMs may produce responses based on outdated, incomplete, or incorrect information, compromising answer quality and reliability. Biases from training data can also lead to unfair or biased responses, requiring evaluation methods that can detect and mitigate such biases. LLMs are sensitive to input phrasing, leading to inconsistencies in responses, which need to be addressed in evaluation methods. Over-optimization can result in plausible but irrelevant answers, necessitating evaluation methods that prioritize relevance and specificity. Finally, verifying the accuracy and reliability of LLM-generated answers is a challenging task, often demanding human intervention, which can be time-consuming and subjective.

Possible future work to address challenges in LLM answer evaluation include developing multi-faceted evaluation metrics that consider various aspects of answer quality, such as relevance, coherence, and factual correctness. Comparing responses generated by multiple LLMs or model versions can help estimate relative answer quality, given that the compared models are reliable and well-performing. Incorporating adversarial training can make LLMs more robust against misleading or ambiguous queries, leading to improved answer generation and evaluation. Integrating active learning and feedback loops in the evaluation process can iteratively improve the model by identifying and correcting errors and inconsistencies. Implementing bias mitigation techniques, such as data augmentation and re-sampling, can reduce biases in LLM-generated responses and enhance fairness. Involving human evaluators as part of the evaluation process can provide valuable insights into the model's performance and aid in refining automated evaluation methods.

3.5. *Using LLM assisted practices in industry*

In the conventional customer enquire and solution process, the initial stage involves manual analysis of the problem statement provided by the customer. Following this, a subject matter expert evaluates the problem and suggests a suitable coating material to resolve the customer's degradation issue. After the expert's assessment, the production manager manually defines the application details, considering the selected coating material and the manufacturing readiness level. Subsequently, the Quality Assurance/Quality Control (QA/QC) department assesses the conformity and quality requirements. The culmination of these steps leads to the final stage where a quotation is drafted and reviewed before being sent to the customer.

On the other hand, the LLM-assisted process introduces automation at various stages, thereby increasing efficiency. Initially, the LLM consumes the problem statement and uses its understanding to suggest an appropriate coating material. This suggestion is then validated or fine-tuned by a human expert. Based on the selected material and indexed data, the LLM proposes a suitable manufacturing method, which is then verified by the production manager. In the next stage, the LLM conducts an automated analysis to identify any potential gaps and check compliance against quality standards, and these findings are then reviewed by a QA/QC expert. The final stage mirrors the conventional process, where an offer to the customer is drafted. However, this process is now faster and more comprehensive due to the inputs from the LLM-assisted stages and expert reviews. Thus, the LLM-assisted process saves substantial time, covers a wider knowledge domain, and allows the team to concentrate more on the implementation of the solution rather than the design stage.

4. Conclusions

In this work, the specific domain GPT-3 model employs an indexing mechanism that helps it access relevant information from the dataset more efficiently. The results demonstrate that by creating an index of the data, the model can quickly retrieve specific information, which results in more accurate and relevant responses. This contrasts with the GPT-4 model, which rely on its pre-trained knowledge and may not have immediate access to the most relevant information for a given query. The fine-tuned data indexed GPT-3 model, benefits from being updated with the most recent and relevant data, providing a more accurate and up-to-date understanding of the target domain. The vector indexing method appears to be superior to the tree method in general query tasks. The addition of the self-evaluation function increases the confidence that the provided answer is correct and relevant. In addition, in this work the source text is provided allowing for easier human intervention when irrelevant sources are highlighted.

In conclusion, this study demonstrates that a fine-tuned data indexed GPT model can significantly improve query response performance compared to state-of-the-art GPT-4. By utilizing domain adaptation and data indexing techniques, this model can provide more accurate, coherent, and relevant responses, which have important implications for the development and application of natural language processing models in surface engineering domains.

CRediT authorship contribution statement

Spyros Kamnis: Conceptualization, Methodology, Data curation, Writing- Original draft preparation, Visualization, Investigation, Software, Validation, Reviewing and Editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Appendix A. Indexing scripts & response evaluation

```
from gpt_index import (
GPTTreeIndex,
GPTSimpleVectorIndex,
SimpleDirectoryReader,
LLMPredictor,
ServiceContext,
Response
)
from langchain.chat_models import ChatOpenAI
from langchain.llms import OpenAI
from gpt_index.evaluation import ResponseEvaluator
import pandas as pd
pd.set_option('display.max_colwidth', 0)

# gpt-3 (davinci)
llm_predictor_gpt3 = LLMPredictor(llm=OpenAI(temperature=0, model_name="text-davinci-003"))
service_context_gpt3 = ServiceContext.from_defaults(llm_predictor=llm_predictor_gpt3)

# gpt-4
llm_predictor_gpt4 = LLMPredictor(llm=ChatOpenAI(temperature=0, model_name="gpt-4"))
service_context_gpt4 = ServiceContext.from_defaults(llm_predictor=llm_predictor_gpt4)

evaluator = ResponseEvaluator(service_context=service_context_gpt3)
evaluator_gpt4 = ResponseEvaluator(service_context=service_context_gpt4)

# Generate tree and vector index
tree_index = GPTTreeIndex.from_documents(documents)
vector_index = GPTSimpleVectorIndex.from_documents(documents)

# Query/Answer/Context Evaluation
def display_eval_df(response: Response, eval_result: str) -> None:
source_text = (
response.source_nodes[0].source_text[:500] + "..."
if response.source_nodes
else "No source available"
)

eval_df = pd.DataFrame(
{
"Response": str(response),
"Source": source_text,
"Evaluation Result": eval_result
},
index=[0]
)

eval_df = eval_df.style.set_properties(
**{
'inline-size': '600px',
'overflow-wrap': 'break-word',
},
subset=["Response", "Source"]
)
display(eval_df)
```

```
#Source Node Evaluation function
from typing import List

# define jupyter display function
def display_eval_sources(query: str, response: Response, eval_result: List[str]) ->
None:

```

```
sources = [s.node.get_text() for s in response.source_nodes]
eval_df = pd.DataFrame(
{
"Source": sources,
"Eval Result": eval_result,
},
)
eval_df.style.set_caption(query)
eval_df = eval_df.style.set_properties(
**{
'inline-size': '600px',
'overflow-wrap': 'break-word',
},
subset=["Source"]
)


display(eval_df)
```

References

[1] T.B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D.M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, GPT-3, NeurIPS, 2020.

[2] L. Floridi, M. Chiriatti, GPT-3: its nature, scope, limits, and consequences, Mind. Mach. (2020), https://doi.org/10.1007/s11023-020-09548-1.

[3] B.D. Lund, T. Wang, Chatting about ChatGPT: how may AI and GPT impact academia and libraries? Libr. Hi Tech News. (2023) https://doi.org/10.1108/LHTN-01-2023-0009.

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Adv. Neural Inf. Process, Syst, 2017, https://doi.org/10.48550/arXiv.1706.03762%0A.

[5] A. Tealab, Time series forecasting using artificial neural networks methodologies: a systematic review, Futur. Comput. Informatics J. (2018), https://doi.org/10.1016/j.fcij.2018.10.003.

[6] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. (1997), https://doi.org/10.1162/neco.1997.9.8.1735.

[7] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C.L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, R. Lowe, Training Language Models to Follow Instructions With Human Feedback, 2022.

[8] J. Liu, LlamaIndex, 2022, https://doi.org/10.5281/zenodo.1234.

[9] M. Tzinava, K. Delibasis, B. Allcock, S. Kamnis, A general-purpose spray coating deposition software simulator, Surf. Coat. Technol. 399 (2020), https://doi.org/10.1016/j.surfcoat.2020.126148.

[10] S. Araguás-Rodríguez, M. Ferran-Marqués, C.C. Pilgrim, S. Kamnis, J.P. Feist, J. R. Nicholls, Thermal history coatings: PART I - influence of atmospheric plasma spray parameters on performance, in: Proc. ASME Turbo Expo, 2020, https://doi.org/10.1115/GT2020-16004.

[11] J. Pulsford, F. Venturi, S. Kamnis, T. Hussain, Sliding wear behaviour of WC-Co reinforced NiCrFeSiB HVOAF thermal spray coatings against WC-Co and Al2O3 counterbodies, Surf. Coat. Technol. 386 (2020), https://doi.org/10.1016/j.surfcoat.2020.125468.

[12] A. Younes, P. Nnamchi, J. Medina, P. Pérez, V.M. Villapún, F. Badimuro, S. Kamnis, E. Jimenez-Melero, S. González, Wear rate at RT and 100 °C and operating temperature range of microalloyed cu<inf>50</inf>Zr<inf>50</inf> shape memory alloy, J. Alloys Compd. 817 (2020), https://doi.org/10.1016/j.jallcom.2019.153330.

[13] M. Tzinava, K. Delibasis, S. Kamnis, Self-organizing maps for optimized robotic trajectory planning applied to surface coating, in: IFIP Adv. Inf. Commun. Technol., 2021, https://doi.org/10.1007/978-3-030-79150-6_16.

[14] S. Kamnis, A.K. Sfikas, B. Allcock, S. Gonzalez, HVOF processed CoCrFeMnNi high-entropy alloy coatings: a combined computational and experimental validation approach, J. Therm. Spray Technol. 31 (2021) 1000–1010, https://doi.org/10.1007/s11666-021-01268-x.

[15] F. Venturi, S. Kamnis, T. Hussain, Internal diameter HVOAF thermal spray of carbon nanotubes reinforced WC-Co composite coatings, Mater. Des. (2021). https://doi.org/https://doi.org/10.1016/j.matdes.2021.109566.

[16] D.K. Iakovidis, M. Ooi, Y.C. Kuang, S. Demidenko, A. Shestakov, V. Sinitsin, M. Henry, A. Sciacchitano, S. Discetti, S. Donati, M. Norgia, A. Menychtas, I. Maglogiannis, S.C. Wriessnegger, L.A.B. Chacon, G. Dimas, D. Filos, A.H. Aletras, J. Töger, F. Dong, S. Ren, A. Uhl, J. Paziewski, J. Geng, F. Fioranelli, R. M. Narayanan, C. Fernandez, C. Stiller, K. Malamousi, S. Kamnis, K. Delibasis, D. Wang, J. Zhang, R.X. Gao, Roadmap on signal processing for next generation measurement systems, Meas. Sci. Technol. (2022), https://doi.org/10.1088/1361-6501/ac2dbd.

[17] A.K. Sfikas, S. Gonzalez, A.G. Lekatou, S. Kamnis, A.E. Karantzalis, A critical review on Al-Co alloys: fabrication routes, microstructural evolution and properties, Metals (Basel) (2022). https://doi.org/https://doi.org/10.3390/met12071092.

[18] A.E. Karantzalis, A. Poulia, S. Kamnis, A. Sfikas, A. Fotsis, E. Georgatis, Modification of cantor high entropy alloy by the addition of Mo and Nb: microstructure evaluation, nanoindentation-based mechanical properties, and sliding wear response assessment, Alloys (2022), https://doi.org/10.3390/alloys1010006.

[19] S. Kamnis, A.K. Sfikas, S. Gonzalez, A.E. Karantzalis, Georgatis, a new cooling-rate-dependent machine learning feature for the design of thermally sprayed high-entropy alloys, J. Therm. Spray Technol. (2022), https://doi.org/10.1007/s11666-022-01462-5.

[20] K. Malamousi, K. Delibasis, B. Allcock, S. Kamnis, Digital transformation of thermal and cold spray processes with emphasis on machine learning, Surf. Coat. Technol. (2022). https://doi.org/https://doi.org/10.1016/j.surfcoat.2022.128138.

[21] K. Dimitriadis, A.K. Sfkas, S. Kamnis, P. Tsolka, S. Agathopoulos, Influence of heat treatment on the microstructure and the physical and mechanical properties of dental highly translucent zirconia, J. Adv. Prosthodont. (2022), https://doi.org/10.4047/jap.2022.14.2.96.

[22] N.H. Faisal, R. Ahmed, N. Sellami, A. Prathuru, J. Njuguna, F. Venturi, T. Hussain, H.Y. Nezhad, N.K. Katiyar, S. Goel, H. Upadhyaya, S. Joshi, F. Muhammad-Sukki, R. Prabhu, T. Mallick, W. Whittow, S. Kamnis, Thermal spray coatings for electromagnetic wave absorption and interference shielding: a review and future challenges, Adv. Eng. Mater. (2022), https://doi.org/10.1002/adem.202200171.

[23] S. González, A.K. Sfikas, S. Kamnis, S.E. John, Z.W. Nye, M. Spink, C. Allen, R. Martínez-Sánchez, S.W. Naung, M. Rahmati, T. Keil, K. Durst, R.J. Lancaster, Computational and experimental investigation of the strain rate sensitivity of small punch testing of the high-entropy alloy CoCrFeMnNi, J. Alloys Compd. (2023), https://doi.org/10.1016/j.jallcom.2022.168219.

[24] S. González, A.K. Sfikas, S. Kamnis, C.G. Garay-Reyes, A. Hurtado-Macias, R. Martínez-Sánchez, Wear resistant CoCrFeMnNi0.8V high entropy alloy with multi length-scale hierarchical microstructure, Mater. Lett. (2023), https://doi.org/10.1016/j.matlet.2022.133504.

[25] J. Pulsford, F. Venturi, Z. Pala, S. Kamnis, T. Hussain, Application of HVOF WC-Co-Cr coatings on the internal surface of small cylinders: effect of internal diameter on the wear resistance, Wear 432–433 (2019), https://doi.org/10.1016/j.wear.2019.202965.

[26] S. Kamnis, K. Malamousi, A. Marrs, B. Allcock, K. Delibasis, Aeroacoustics and artificial neural network modeling of airborne acoustic emissions during high kinetic energy thermal spraying, J. Therm. Spray Technol. 28 (2019), https://doi.org/10.1007/s11666-019-00874-0.

[27] V. Katranidis, S. Kamnis, B. Allcock, S. Gu, Effects and interplays of spray angle and stand-off distance on the sliding wear behavior of HVOF WC-17Co coatings, J. Therm. Spray Technol. 28 (2019), https://doi.org/10.1007/s11666-019-00831-x.

[28] V. Katranidis, S. Kamnis, S. Gu, Prediction of coating properties of thermally sprayed WC–Co on complex geometries, J. Therm. Spray Technol. 27 (2018), https://doi.org/10.1007/s11666-018-0739-6.

[29] J. Pulsford, S. Kamnis, J. Murray, M. Bai, T. Hussain, Effect of particle and carbide grain sizes on a HVOAF WC-Co-Cr coating for the future application on internal surfaces: microstructure and wear, in: Proc. Int. Therm. Spray Conf., 2017.

[30] V. Katranidis, S. Gu, D.C. Cox, M.J. Whiting, S. Kamnis, FIB-SEM sectioning study of decarburization products in the microstructure of HVOF-sprayed WC-Co coatings, J. Therm. Spray Technol. 27 (2018), https://doi.org/10.1007/s11666-018-0721-3.

[31] V. Katranidis, S. Gu, T.R. Reina, E. Alpay, B. Allcock, S. Kamnis, Experimental study of high velocity oxy-fuel sprayed WC-17Co coatings applied on complex geometries. Part B: influence of kinematic spray parameters on microstructure, phase composition and decarburization of the coatings, Surf. Coat. Technol. 328 (2017), https://doi.org/10.1016/j.surfcoat.2017.09.027.

[32] V. Katranidis, S. Gu, B. Allcock, S. Kamnis, Experimental study of high velocity oxy-fuel sprayed WC-17Co coatings applied on complex geometries. Part a: influence of kinematic spray parameters on thickness, porosity, residual stresses and microhardness, Surf. Coat. Technol. 311 (2017), https://doi.org/10.1016/j.surfcoat.2017.01.015.

[33] M. Mahrukh, A. Kumar, S. Gu, S. Kamnis, Computational development of a novel aerosol synthesis technique for production of dense and nanostructured zirconia coating, Ind. Eng. Chem. Res. 55 (2016), https://doi.org/10.1021/acs.iecr.6b01725.

[34] M. Mahrukh, A. Kumar, S. Gu, S. Kamnis, E. Gozali, Modeling the effects of concentration of solid nanoparticles in liquid feedstock injection on high-velocity suspension flame spray process, Ind. Eng. Chem. Res. 55 (2016), https://doi.org/10.1021/acs.iecr.5b03956.

[35] E. Gozali, M. Mahrukh, S. Gu, S. Kamnis, Numerical investigation on effects of nanoparticles on liquid feedstock behavior in high velocity oxygen fuel (HVOF) suspension spraying, Surf. Coat. Technol. 280 (2015), https://doi.org/10.1016/j.surfcoat.2015.09.012.

[36] Z. Zhu, S. Kamnis, S. Gu, Numerical study of molten and semi-molten ceramic impingement by using coupled Eulerian and Lagrangian method, Acta Mater. 90 (2015), https://doi.org/10.1016/j.actamat.2015.02.010.

[37] E. Gozali, M. Mahrukh, S. Gu, S. Kamnis, Numerical analysis of multicomponent suspension droplets in high-velocity flame spray process, J. Therm. Spray Technol. 23 (2014), https://doi.org/10.1007/s11666-014-0106-1.

[38] E. Gozali, S. Kamnis, S. Gu, Numerical investigation of combustion and liquid feedstock in high velocity suspension flame spraying process, Surf. Coat. Technol. 228 (2013), https://doi.org/10.1016/j.surfcoat.2013.04.026.
[39] A. Kumar, S. Gu, H. Tabbara, S. Kamnis, Study of impingement of hollow ZrO2 droplets onto a substrate, Surf. Coat. Technol. 220 (2013), https://doi.org/10.1016/j.surfcoat.2012.08.061.
[40] A. Kumar, S. Gu, S. Kamnis, Simulation of impact of a hollow droplet on a flat surface, Appl. Phys. A Mater. Sci. Process. 109 (2012), https://doi.org/10.1007/s00339-012-7043-y.
[41] S. Kamnis, S. Gu, M. Vardavoulias, Numerical study to examine the effect of porosity on in-flight particle dynamics, J. Therm. Spray Technol. 20 (2011), https://doi.org/10.1007/s11666-010-9606-9.
[42] S. Kamnis, S. Gu, M. Vardavoulias, The influence of powder porosity on the bonding mechanism at the impact of thermally sprayed solid particles, Metall. Mater. Trans. A Phys. Metall. Mater. Sci. 41 (2010), https://doi.org/10.1007/s11661-010-0488-8.
[43] S. Gu, S. Kamnis, Bonding mechanism from the impact of thermally sprayed solid particles, Metall. Mater. Trans. A Phys. Metall. Mater. Sci. 40 (2009), https://doi.org/10.1007/s11661-009-9959-1.
[44] S. Gu, S. Kamnis, Numerical modelling of in-flight particle dynamics of non-spherical powder, Surf. Coat. Technol. 203 (2009), https://doi.org/10.1016/j.surfcoat.2009.05.024.
[45] S. Kamnis, S. Gu, T.J. Lu, C. Chen, Numerical modeling the bonding mechanism of HVOF sprayed particles, Comput. Mater. Sci. 46 (2009), https://doi.org/10.1016/j.commatsci.2009.05.009.
[46] S. Kamnis, S. Gu, Study of in-flight and impact dynamics of nonspherical particles from HVOF guns, J. Therm. Spray Technol. 19 (2010), https://doi.org/10.1007/s11666-009-9382-6.
[47] S. Kamnis, S. Gu, T.J. Lu, C. Chen, Numerical modelling of sequential droplet impingements, J. Phys. D. Appl. Phys. 41 (2008), https://doi.org/10.1088/0022-3727/41/16/165303.
[48] N. Zeoli, S. Gu, S. Kamnis, Numerical simulation of in-flight particle oxidation during thermal spraying, Comput. Chem. Eng. 32 (2008), https://doi.org/10.1016/j.compchemeng.2007.08.008.
[49] S. Kamnis, S. Gu, T.J. Lu, C. Chen, Computational simulation of thermally sprayed WC-Co powder, Comput. Mater. Sci. 43 (2008), https://doi.org/10.1016/j.commatsci.2008.03.015.
[50] S. Kamnis, S. Gu, N. Zeoli, Mathematical modelling of Inconel 718 particles in HVOF thermal spraying, Surf. Coat. Technol. 202 (2008), https://doi.org/10.1016/j.surfcoat.2007.10.006.
[51] S. Kamnis, S. Gu, Computational fluid dynamic modelling of water-cooling mechanism during thermal spraying process, Int. J. Model. Identif. Control. 2 (2007), https://doi.org/10.1504/IJMIC.2007.014940.
[52] S. Kamnis, S. Gu, Numerical modelling of propane combustion in a high velocity oxygen-fuel thermal spray gun, Chem. Eng. Process. Process Intensif. 45 (2006), https://doi.org/10.1016/j.cep.2005.06.011.
[53] S. Kamnis, S. Gu, 3-D modelling of kerosene-fuelled HVOF thermal spray gun, Chem. Eng. Sci. 61 (2006), https://doi.org/10.1016/j.ces.2006.04.005.
[54] S. Kamnis, S. Gu, Numerical modelling of droplet impingement, J. Phys. D. Appl. Phys. 38 (2005), https://doi.org/10.1088/0022-3727/38/19/015.
[55] H. Seetha, V. Tiwari, K.R. Anugu, D.S. Makka, D.R. Karnati, A GUI based application for PDF processing tools using python & CustomTkinter, Int. J. Res. Appl. Sci. Eng. Technol. (2023), https://doi.org/10.22214/ijraset.2023.48848.
[56] A. Farkiya, P. Saini, S. Sinha, S. Desai, Natural language processing using NLTK and WordNet, Int. J. Comput. Sci. Inf. Technol. 6 (2015) 5465–5469.
[57] ECMA-404, The JSON Data Interchange Format, ECMA Int., 2013.