

Projektowanie belki wspornikowej z wykorzystaniem sztucznej inteligencji

Cantilever beam design using Artificial Intelligence



dr hab. inż. Anna Derlatka

Politechnika Częstochowska
anna.derlatka@pcz.pl
ORCID: 0000-0002-6509-2706



prof. dr hab. inż. Piotr Lacki

Politechnika Częstochowska
piotr.lacki@pcz.pl
ORCID: 0000-0002-0787-8890

Streszczenie

Celem pracy była rozbudowa systemu rozwiązującego zadanie inżynierskie z wykorzystaniem języka naturalnego. Przykład opisany w artykule przedstawia rozwiązanie problemu belki wspornikowej. Rozwijany system DuNLP wykorzystał język naturalny do tworzenia kodu systemu ADINA za pomocą modelu generacji języka. Niektóre parametry (dotyczące np. właściwości materiału belki) zostały użyte z wewnętrznej bazy danych, a pozostałe (m.in. kształt przekroju poprzecznego belki) wymagały szkolenia sztucznej inteligencji.

W szczególności proces ten miał na celu nauczenie jej podstawowych poleceń kodu ADINA. DuNLP umożliwia rozwiązywanie problemów technicznych osobom bez specjalistycznej wiedzy inżynierskiej lub numerycznej, jednocześnie ograniczając wymagane od użytkownika umiejętności programistyczne. DuNLP to koncepcja, która może być stosowana do rozwiązywania różnych zagadnień inżynierii konstrukcyjnej przy użyciu szerokiej gamy oprogramowania wykorzystującego pliki tekstowe zawierające listę poleceń.

Słowa kluczowe sztuczna inteligencja, przetwarzanie języka naturalnego, metoda elementów skończonych, belka stalowa

Abstract

The goal of this work was to develop a system that solves engineering problems using natural language. The example presented in this paper solves the cantilever beam problem. The developed DuNLP system used natural language to generate ADINA code using the language generation model. Some parameters (such as the beam's material properties) were used from the internal database. Other parameters (including the beam's cross-sectional shape) required training

of the AI. Specifically, the training aimed to teach the AI the basic ADINA code commands. DuNLP enables people without specialised engineering or numerical knowledge to solve engineering problems and reduce the required programming skills. DuNLP is a concept that can be used to solve various structural engineering problems, leveraging a wide range of software that processes text files containing command lists.

Keywords artificial intelligence, natural language processing, finite element method, steel beam

WPROWADZENIE

Do przewidywania zachowania konstrukcji budowlanych wykorzystuje się m.in. oprogramowanie do modelowania MES, takie jak Abaqus i ANSYS. Program ADINA [1], zastosowany w niniejszej pracy, również należy do tej grupy narzędzi. Służy on do rozwiązywania najtrudniejszych problemów nieliniowych, obejmujących nieliniowości geometryczne, materiałowe i obciążeniowe, duże odkształcenia oraz warunki kontak-

towe. Interfejs użytkownika ADINA (AUI) zapewnia pełne możliwości wstępnego i końcowego przetwarzania wszystkich modułów rozwiązania, co pozwala użytkownikom na ukończenie całego procesu symulacji. Ograniczeniem programów do modelowania MES jest konieczność znajomości tworzenia modelu w oparciu o interfejs programu lub skrypt. Ich niewątpliwą zaletą jest łatwość obsługi przez użytkowników. Niemniej jednak wymagają one wiedzy inżynierskiej.

Podejścia oparte na uczeniu maszynowym (ang. Machine Learning – ML) i sztucznej inteligencji (ang. Artificial Intelligence – AI) znajdują zastosowanie jako narzędzia do projektowania materiałów [2] i konstrukcji dla inżynierii lądowej. Dostępność algorytmów ML umożliwiła wprowadzenie wielu innowacji m.in. w zakresie betonu, obejmujących opracowywanie składu, przewidywanie jego właściwości [3–5] lub analizę nośności konstrukcji żelbetowej [6].



Coraz częściej sztuczne sieci neuronowe (ang. Artificial Neural Networks – ANN) są stosowane do przewidywania nośności różnych stalowych elementów konstrukcyjnych, takich jak: belki stalowe [7], dźwigary ze środknikiem z blachy trapezowej [8] oraz zimnogięte okrągłe kształtowniki zamknięte [9]. Niemniej jednak sztuczne sieci neuronowe mogą być używane jako narzędzie do rozwiązywania złożonych problemów. Autorzy pracy [10] zastosowali ANN jako narzędzie do identyfikacji głębokości pęknięć w stalowych konstrukcjach belek w oparciu o analizę drgań. W pracy [11] przyjęto metodę k-średnich do określenia właściwości mechanicznych kompozytowych prętów zbrojonych z wykorzystaniem podejścia mikromechanicznego, a także analizy postępujących procesów uszkodzeń występujących pod wpływem obciążenia zginającego.

Sztuczna inteligencja znajduje również zastosowanie w konstrukcjach zespolonych. Autorzy prac [12, 13] uznali, że zgodność i niezawodność opracowanych modeli ANN była wystarczająca do przewidywania zachowania konstrukcji zespolonych stalowo-betonowych. Nato-

miast w pracy [14] przedstawiono wyniki analizowanych modeli ANN i algorytmu genetycznego (ang. Genetic Algorithm – GA) do optymalizacji zespolonej belki stalowo-drewnianej.

Typowy proces uczenia maszynowego składa się z następujących etapów:

- zdefiniowanie problemu;
- przeprowadzenie badania eksperymentalnego na próbce referencyjnej;
- opracowanie i walidacja modelu numerycznego;
- wykonanie obliczeń numerycznych w celu zebrania danych dla modelu ML;
- opracowanie, ewaluacja i wdrożenie modelu ML.

Największym wyzwaniem jest opracowanie modelu numerycznego i zebranie danych. Wymagają one wiedzy inżynierskiej i znajomości poleceń programu do modelowania MES. Niezbędna jest również umiejętność budowania sieci neuronowych. W tym zakresie pomocne mogą być systemy oparte na przetwarzaniu języka naturalnego (ang. Natural Language Processing – NLP).

Modele generacji języka (ang. Large Language Model – LLM), trenowane

na dużym zbiorze danych tekstowych, są w stanie zrozumieć kontekst rozmowy. Jak przedstawiono w pracy [15], LLM wykazał potężne możliwości w zakresie interakcji z ludźmi. W artykułach [16, 17] wskazano, że LLM może stanowić cenne narzędzie dla osób, dla których angielski nie jest językiem ojczystym, wspomagające doskonalenie umiejętności pisania i naukę programowania. Próby wykorzystania zapytań języka naturalnego przeprowadzono w następujących dziedzinach inżynierii lądowej: pobieranie podzbiorów modeli BIM w projektach budowlanych [18], zarządzanie energią w budynkach [19] i harmonogramowanie projektów budowlanych [20]. Jednak ich wyniki oceniono jako niewystarczające. Przeprowadzono również krytyczną ocenę modeli generacji języka [21, 22], podkreślając trudności z liczeniem. Wynika to z ogólnego zrozumienia tematu, ponieważ szkolenie modeli LLM opiera się na nauce z szerokiej gamy tekstów pochodzących głównie z internetu. Warto przy tym zaznaczyć, że istnieje możliwość trenowania modeli LLM przez użytkownika z wykorzystaniem konkretnych danych.

CEL PRACY

Niniejsze opracowanie przedstawia możliwość zastosowania modelu LLM w projektowaniu konstrukcji inżynierskich. Wykorzystuje ono najnowsze osiągnięcia w dziedzinie czatów opartych na sztucznej inteligencji, co pozwala na uproszczenie procesu projektowania poprzez wydawanie poleceń w języku naturalnym. Do analizy posłużono się zasobami bazy danych AI dotyczącymi właściwości materiałów i geometrii przekrojów poprzecznych belek stalowych. Zakres pracy obejmował nie tylko weryfikację wiedzy modelu LLM, ale także jego nauczanie.

Założeniem autorów było rozwijanie systemu (ang. Design Using Natural Language Processing – DuNLP) rozwiązującego zadanie inżynierskie z wykorzystaniem języka naturalnego. Jego działanie przedstawiono na przykładzie belki wspornikowej. Dzięki opracowaniu bliźniaka cyfrowego belkę stalową można przeanalizować za pomocą modelu LLM, programu ADINA, sztucznej sieci neuronowej i algorytmu genetycznego. Użytkownik ustawia parametry belki w LLM, po czym generowany jest kod dla programu ADINA, który następnie rozwiązuje problem belki. W końcowym etapie otrzymuje się wyniki, takie jak przemieszczenia lub rozkład naprężeń w belce.

ZAKRES PRACY

Na początku sprawdzono znajomość wewnętrznych baz danych modelu LLM. Weryfikacja polegała na odpytywaniu LLM. Poziom wiedzy z zakresu właściwości materiałów, wymiarów przekrojów stalowych oraz poleceń kodu ADINA oceniono odpowiednio na 60, 30 i 1%.

Następnie model LLM został nauczony i sprawdzony pod kątem znajomości poleceń kodu ADINA oraz wewnętrznych baz danych, głównie w zakresie danych materiałowych i wymiarów przekrojów stalowych. W dalszej kolejności przeprowadzono weryfikację nabytej wiedzy. Baza danych została zweryfikowana poprzez odpytywanie LLM. W przypadku błędów model LLM uczył się prawidłowych war-

tości. Czynności te powtarzano do momentu osiągnięcia 100% poziomu wiedzy we wszystkich analizowanych obszarach.

Duży nacisk położono na szkolenie i testowanie wymiarów przekrojów stalowych, co polegało na nauczaniu modelu LLM nazw, symboli i wartości głównych parametrów profili stalowych IPE. LLM został nauczony następujących wielkości geometrycznych: wysokości całkowitej (h), szerokości pasa (b), grubości pasa (t_f), grubości środnika (t_w) i promienia zaokrąglenia (r) każdego profilu w zakresie od IPE 80 do IPE 600.

Duży nacisk położono na testowanie wymiarów przekrojów stalowych IPE, ucząc model LLM nazw, symboli i głównych parametrów.

Następnie zdobytą wiedzę zweryfikowano, sprawdzając wszystkie parametry wybranego przekroju poprzecznego lub pojedyncze parametry konkretnego przekroju.

Na koniec użytkownik mógł zdefiniować problem i odpytać LLM za pomocą języka naturalnego. W odpowiedzi wygenerowano polecenia w kodzie ADINA, a następnie użytkownik przeprowadził obliczenia numeryczne w programie ADINA.

Model LLM został użyty do zastąpienia języka naturalnego kodem programu

do wygenerowania kodu belki wspornikowej w ADINA:

„Wygeneruj kod w interfejsie użytkownika ADINA (AUI), definiując linię o długości 3,22 m, a warunek brzegowy w punkcie 2 blokuje wszystkie stopnie swobody. Zdefiniuj obciążenie ciężarem własnym. Zdefiniuj siłę $F = 1,45$ kN i przyłóż ją w punkcie 1. Zdefiniuj właściwości mechaniczne materiału S235 i przekrój poprzeczny belki IPE 160 wraz z jego parametrami. Dodaj definicję grupy elementów belki. Przypisz podział siatki do całej

bieżącej geometrii modelu za pomocą metody LENGTH co 10 mm. Wygeneruj elementy belki w linii 1”.

Belka wspornikowa wygenerowana za pomocą kodu LLM oraz obliczona z wykorzystaniem programu ADINA została przedstawiona na rysunku. Uznano, że kod do programu ADINA, wygenerowany przez LLM, uwzględnia właściwą długość belki, wymiary przekroju IPE 160, właściwości materiału oraz warunki brzegowe i obciążenia. Po przeliczeniu belki

Opracowana koncepcja DuNLP może znacząco wpłynąć na uproszczenie i przyspieszenie procesu projektowania konstrukcji budowlanych.

ADINA. Niektóre parametry (takie jak właściwości materiału belki) zostały wykorzystane z wewnętrznej bazy danych LLM. Pozostałe, m.in. kształt przekroju poprzecznego belki, wymagały przeszkolenia sztucznej inteligencji. W szczególności miało ono na celu nauczanie jej podstawowych poleceń kodu ADINA.

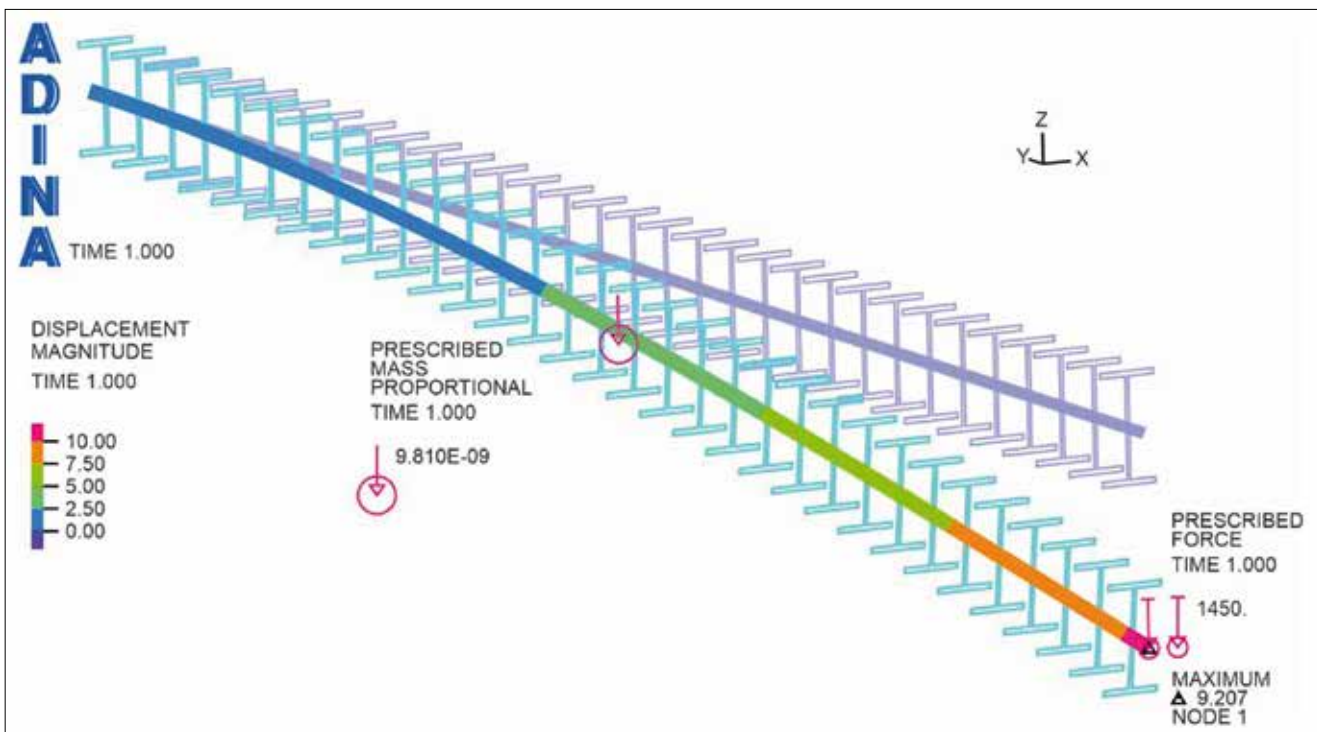
WYNIKI

Poniżej przedstawiono przykład użycia modelu przetwarzania języka naturalnego

program wskazał jej ugięcie równe 9,207 mm, które uznano za właściwe (rysunek).

DYSKUSJA

Rozwijany projekt DuNLP pozwala na wygenerowanie w pełni sparametryzowanego kodu skryptu przeznaczonego do przeprowadzania obliczeń belki. Dzięki zastosowaniu parametrycznego modelu numerycznego oraz możliwości uruchamiania programu ADINA w trybie wsadowym, użytkownik zyskuje możliwość



Rys. Belka wspornikowa wygenerowana za pomocą koncepcji DuNLP wraz z zaznaczonymi przemieszczeniami

automatycznego generowania i analizowania bardzo dużej liczby wariantów obliczeniowych. W praktyce oznacza to możliwość uzyskania niemal nieograniczonej liczby rozwiązań dla różnych kombinacji parametrów wejściowych, takich jak geometria, właściwości materiałowe lub warunki brzegowe.

Opracowana koncepcja DuNLP może znacząco wpłynąć na uproszczenie i przyspieszenie procesu projektowania konstrukcji budowlanych poprzez umożliwienie

blemów związanych z utworzeniem bazy danych, a następnie nauczeniem jej wszystkich możliwych wariantów działań matematycznych na liczbach. W odniesieniu do analizowanej belki wygenerowanie bazy danych ze wszystkimi wariantami parametrów i wynikami, w celu nauczenia sztucznej inteligencji rozwiązywania problemu, jest praktycznie niemożliwe.

Dlatego kluczową zaletą przedstawionej koncepcji DuNLP jest zdolność nauczenia skryptu systemu ADINA poleceń,

wia projektowanie konstrukcji przez osoby nieposiadające specjalistycznej wiedzy inżynierskiej. Dodatkowo użytkownik nie musi posiadać zaawansowanych umiejętności programistycznych ani wiedzy z zakresu metod numerycznych. Zapewnia to metoda komunikacji między użytkownikiem a modelem LLM, która gwarantuje spersonalizowaną, zbliżoną do ludzkiej interakcję z wykorzystaniem języka naturalnego.

Opracowana koncepcja ma charakter uniwersalny i może zostać z powodzeniem zastosowana również w innych zagadnieniach obliczeniowych realizowanych w środowisku ADINA. Pozwala to na rozszerzenie jej użycia na bardziej złożone konstrukcje i różnorodne problemy inżynierskie, co czyni ją narzędziem o dużym potencjale praktycznym i badawczym.

Istotnym elementem proponowanego podejścia jest tworzenie bazy danych zawierającej zestawy parametrów wejściowych oraz odpowiadające im wyniki obliczeń. Taka baza danych stanowi cenne źródło informacji, które może zostać wykorzystane do trenowania modeli sztucznej inteligencji, w szczególności sztucznych sieci neuronowych (ANN). W efekcie możliwe jest

Zaletą DuNLP jest uczenie poleceń systemu ADINA i generowanie przez sztuczną inteligencję danych do pliku wstępnego przetwarzania.

wydawania poleceń w języku naturalnym. Takie podejście eliminuje konieczność ręcznego tworzenia skomplikowanych modeli numerycznych lub pisania zaawansowanych skryptów, wymagających specjalistycznej wiedzy z zakresu programowania oraz obsługi oprogramowania budowlanego.

Z inżynierskiego punktu widzenia największą wadą sztucznej inteligencji są trudności z liczeniem. Wynikają one z pro-

dzięki czemu sztuczna inteligencja generuje dokładne dane do pliku do wstępnego przetwarzania. Następnie program ADINA wykorzystuje ten plik, wykonuje obliczenia przy użyciu metody elementów skończonych i generuje plik końcowego przetwarzania. Dzięki temu podejściu użytkownik otrzymuje dokładne wyniki analizowanej konstrukcji. Co więcej, należy podkreślić, że podejście DuNLP umożli-

opracowanie tzw. cyfrowego bliźniaka analizowanego układu, który odwzorowuje jego zachowanie w sposób szybki i efektywny.

Wytrenowana sieć neuronowa jest w stanie przewidywać rozwiązania dla zadanych parametrów znacznie szybciej niż klasyczne metody numeryczne, co w wielu przypadkach można interpretować jako działanie w czasie zbliżonym do rzeczywistego. Takie podejście otwiera nowe możliwości w zakresie wspomagania projektowania, analizy wariantowej oraz podejmowania decyzji inżynierskich.

Ponadto wytrenowana sieć neuronowa może zostać wykorzystana jako narzędzie wspierające proces optymalizacji. Jednym z kluczowych problemów inżynierskich jest bowiem znalezienie rozwiązania optymalnego przy zadanych kryteriach. W tym kontekście szczególnie efektywne okazują się metody inspirowane naturą, takie jak algorytmy genetyczne, które umożliwiają przeszukiwanie dużych przestrzeni rozwiązań. Połączenie algorytmów genetycznych z szybkim modelem zastępczym w postaci sieci neuronowej pozwala znacząco skrócić czas obliczeń, jednocześnie zachowując wysoką jakość uzyskiwanych wyników.

WNIOSKI

Koncepcja DuNLP upraszcza proces projektowania konstrukcji budowlanych poprzez zadawanie zapytań w języku naturalnym. Do wykonania projektu elementu konstrukcyjnego można wykorzystać dane już istniejące w modelu LLM, np. właściwości materiałów lub wymiary przekroju. Jednocześnie możliwe jest rozszerzanie baz danych LLM poprzez uczenie sztucznej inteligencji przez użytkownika. W DuNLP największy nacisk położono na trenowanie poleceń systemu ADINA, ponieważ początkowo sztuczna inteligencja znała je w bardzo ograniczonym zakresie. Ostatecznie LLM został wyszkolony do tego stopnia, że potrafił generować sparametryzowany kod skryptu systemu ADINA do projektowania belki stalowej. Kluczową zaletą DuNLP jest zatem możliwość uczenia poleceń skryptu ADINA, dzięki czemu sztuczna inteligencja tworzy dokładne

dane do pliku wstępnego przetwarzania. Jednak do wygenerowania pliku końcowego wymagana jest licencja oprogramowania tego systemu.

Należy podkreślić, że DuNLP umożliwia rozwiązywanie problemów technicznych osobom bez specjalistycznej wiedzy inżynierskiej lub numerycznej, jednocześnie ograniczając wymagane od użytkownika umiejętności programistyczne. To koncepcja, która może być wykorzystywana do rozwiązywania różnych problemów inżynierii konstrukcyjnej przy użyciu szerokiej gamy oprogramowania obsługującego pliki tekstowe zawierające listy poleceń. ■

Literatura

- [1] ADINA System Online Manuals. ADINA R&D, Inc., 2020.
- [2] Lacki P. i in., „Topology Optimization of Gyroid-Based Mechanical Metamaterials Using Artificial Intelligence” w: Iványi P., Krus J., Topping B. (red.), *Proceedings of the 18th International Conference on Civil, Structural and Environmental Engineering Computing*, Cagliari, Sardinia, Italy, 2025, s. 1–10, DOI: 10.4203/cc.10.3.3.
- [3] Li Z. i in., „Machine learning in concrete science: applications, challenges, and best practices” w: *npj Computational Materials*, nr 8, 2022, DOI: 10.1038/s41524-022-00810-x.
- [4] Ben Chaabene W., Flah M., Nehdi M.L., „Machine learning prediction of mechanical properties of concrete: Critical review” w: *Construction and Building Materials*, nr 260, 2020, s. 119889, DOI: 10.1016/j.conbuildmat.2020.119889.
- [5] Fernandes R. i in., „Smart modelling system for alkali-activated concrete pavements using machine learning techniques” w: *Asian Journal of Civil Engineering*, nr 24, 2023, s. 2193–2213, DOI: 10.1007/s42107-023-00635-z.
- [6] Lacki P. i in., „Digital Twin of the Reinforced Concrete Slab Based on the Artificial Neural Network” w: Iványi P., Krus J., Topping B. (red.), *Proceedings of the 7th International Conference on Artificial Intelligence, Soft Computing, Machine Learning and Optimization in Engineering*, Cagliari, Sardinia, Italy, 2025, s. 1–10, DOI: 10.4203/cc.11.1.1.
- [7] Gawande G.S., Gupta L.M., „Rotation Capacity Prediction of Open Web Steel Beams Using Artificial Neural Networks” w: *International Journal of Steel Structures*, nr 4, 2023, DOI: 10.1007/s13296-023-00750-2.
- [8] Kumar S.A., Sofi F.A., Bhat J.A., „Estimation of patch-loading resistance of steel girders with unequal trapezoidal web-corrugation folds using nonlinear FE models and artificial neural networks” w: *Structures*, nr 48, 2023, s. 1651–1669, DOI: 10.1016/j.jistruc.2023.01.049.
- [9] Rabi M. i in., „Prediction of the cross-sectional capacity of cold-formed CHS using numerical modelling and machine learning” w: *Results in Engineering*, nr 17, 2023, s. 100902, DOI: 10.1016/j.rineng.2023.100902.
- [10] Khatir A. i in., „Vibration-based crack prediction on a beam model using hybrid butterfly optimization algorithm with artificial neural network” w: *Frontiers of Structural and Civil Engineering*, nr 16, 2022, s. 976–989, DOI: 10.1007/s11709-022-0840-2.
- [11] Lesiuk G. i in., „Investigation of flexural behaviour of composite rebars for concrete reinforcement with experimental, numerical and machine learning approaches” w: *Philosophical Transactions. Series A, Mathematical, Physical, and Engineering Sciences*, nr 381, 2023, s. 20220394, DOI: 10.1098/rsta.2022.0394.
- [12] Hosseinpour M. i in., „Neural networks-based formulation for predicting ultimate strength of bolted shear connectors in composite cold-formed steel beams” w: *Engineering Applications of Artificial Intelligence*, t. 118, 2023, s. 105614, DOI: 10.1016/j.engappai.2022.105614.
- [13] Hosseinpour M. i in., „New predictive equations for LDB strength assessment of steel-concrete composite beams” w: *Engineering Structures*, t. 258, 2022, s. 114121, DOI: 10.1016/j.engstruct.2022.114121.
- [14] Derlatka A. i in., „Optimization of a steel-timber composite beam using AI” w: *Advances in Steel and Steel-Concrete Composite Structures*, 2025, s. 246–253, DOI: 10.1201/9781003534464-23.
- [15] Kannis S., „Generative pre-trained transformers (GPT) for surface engineering” w: *Surface and Coatings Technology*, t. 466, 2023, s. 129680, DOI: 10.1016/j.surfcoat.2023.129680.
- [16] Pérez-Mercado R. i in., „ChatbotSQL: Conversational agent to support relational database query language learning” w: *SoftwareX*, t. 22, 2023, s. 101346, DOI: 10.1016/j.softx.2023.101346.
- [17] Lecler A., Duron L., Soyer P., „Revolutionizing radiology with GPT-based models: Current applications, future possibilities and limitations of ChatGPT” w: *Diagnostic and Interventional Imaging*, t. 104, nr 6, 2023, s. 269–274, DOI: 10.1016/j.diii.2023.02.003.
- [18] Yin M. i in., „An ontology-aided, natural language-based approach for multi-constraint BIM model querying” w: *Journal of Building Engineering*, t. 76, 2023, s. 107066, DOI: 10.1016/j.jobe.2023.107066.
- [19] Zhang C., Lu J., Zhao Y., „Generative pre-trained transformers (GPT)-based automated data mining for building energy management: Advantages, limitations and the future” w: *Energy and Built Environment*, t. 5, 2023, s. 143–169, DOI: 10.1016/j.enbenv.2023.06.005.
- [20] Prieto S.A., Mengiste E.T., García de Soto B., „Investigating the Use of ChatGPT for the Scheduling of Construction Projects” w: *Buildings*, t. 13, 2023, s. 857, DOI: 10.3390/buildings13040857.
- [21] Lee M., „A Mathematical Investigation of Hallucination and Creativity in GPT Models” w: *Mathematics*, t. 11, nr 10, 2023, s. 2320, DOI: 10.3390/math11102320.
- [22] Tafferner Z. i in., „Can ChatGPT Help in Electronics Research and Development? A Case Study with Applied Sensors” w: *Sensors*, t. 23, nr 10, 2023, s. 4879, DOI: 10.3390/s23104879.