

Чеботарьова Анна Владиславівна, здобувач вищої освіти Дніпровського національного університету імені Олеся Гончара, Україна

Науковий керівник: Сидорова Марина Геннадіївна, кандидат технічних наук, доцент кафедри математичного забезпечення ЕОМ Дніпровського національного університету імені Олеся Гончара, Україна

ГЕНЕРАЦІЯ МУЗИКИ З УРАХУВАННЯМ ЕМОЦІЙНОЇ СКЛАДОВОЇ КОРИСТУВАЦЬКОГО ЗАПИТУ

Анотація. Штучний інтелект посідає важливе місце в житті людей та бізнесі. З його допомогою можна значно пришвидшити процеси та розробити економічно вигідні рішення. Все більше компаній використовують згенеровану музику замість музики, створеною людиною, оскільки це швидкий та дешевий підхід. Велика кількість генераторів музики пропонує користувачу обрати багато музичних параметрів для генерації музики, вплив яких розуміє тільки професійний музикант. Представлений підхід пропонує генерувати музику на основі емоції, яку користувач хоче почути у музиці. Це дозволяє зробити процес генерації музики більш простішим та швидшим для користувачів, які не мають музичної освіти або музичного досвіду.

КЛЮЧОВІ СЛОВА: Нейронні мережі, Генеративні змагальні нейронні мережі (GAN), Класифікація тексту за емоційною складовою, NLP.

Abstract. Artificial Intelligence occupies an important place in people's lives and business. With AI help people can significantly speed up processes and develop cost-effective solutions. More companies are using generated music instead of human-made music because it is a quick and cheaper approach. Most music generators offer the user to choose many musical parameters, the effect of which is understood only by a professional musician. The presented approach proposes to generate music based on the emotion that the user wants to hear in the generated music. This makes the process of music generation easier and faster for users with no musical education or musical experience.

KEY WORDS: Neural networks, Generative adversarial neural networks (GAN), Text classification by emotions, NLP.

Вступ. Генерація видів мистецтва, таких як музика, зображення, текст – є однією з найскладніших завдань штучного інтелекту. Багато існуючих архітектур можуть тільки частково згенерувати музику, зображення чи текст, які б містили такі складні для математичних алгоритмів, але прості для людського розуміння речі як емоцію, підтекст, порівняння, тощо. Найуспішнішою серед таких моделей є генеративні змагальні нейронні мережі (GAN) [1]. На відміну від інших моделей штучного інтелекту, вони можуть генерувати складні в машинному розумінні речі, що призвело до їх широкого використання State-of-the-art [2] напрямку. Через емоції людина може описати багато складних та незрозумілих речей, в своєму роді емоції є інтуїтивним засобом комунікації між людьми, але комунікація між

людьми та машинами все ще залишається дуже складною в деяких аспектах. Одним з таких аспектів є генерація музики через програми-генератори, де користувачу потрібно обрати багато комплексних параметрів, що робить генерацію музики дуже складним процесом з людської точки зору. Представлений проект пропонує спростити генерацію музики шляхом заміни параметрів генерації на описове речення, яке містить емоцію [3], яку користувач хоче почути у згенерованій музиці. Такий підхід підійде користувачам, яким потрібно викликати згенерованою музикою емоції у людей, збільшивши їх досвід від прослуховування. Наприклад, це можуть використовувати у розробці ігор, де в конкретний момент часу потрібно викликати у гравця необхідну емоцію, в презентаціях

художніх виставок, театральних виставах, як музичний супровід у відео. Використання такого підходу є економічно вигіднішим для генерації нескладної, з музичної точки зору мелодії, оскільки не потребує наймати музикантів та ансамблі.

Основна частина. У цій роботі розроблено програмне забезпечення, що вирішує описану вище задачу та з двох частин. Перша частина – комплекс підходів для класифікації користувацького запиту за класом емоції. Друга частина – генерація музики у вигляді спектрограм та подальше перетворення згенерованих спектрограм у музичну послідовність формату .wav.

Загальний алгоритм полягає у тому, що користувач вводить речення (запит), що класифікується за емоцією (любов, злість, здивування, радість, страх, смуток) та співвідноситься з жанром музики, за яким і буде здійснюватися генерування нової композиції (рис.1.).

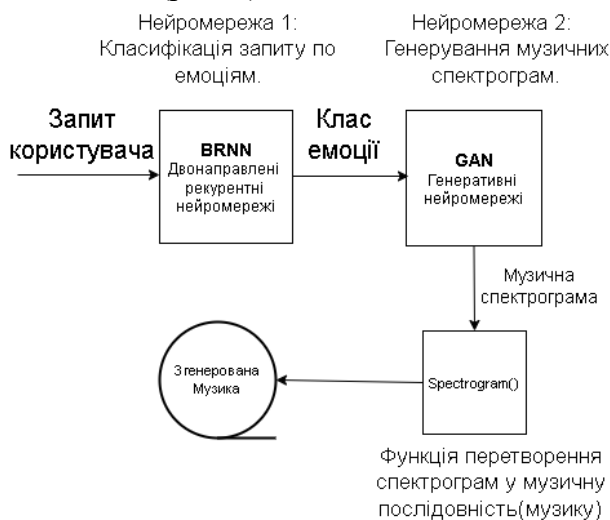


Рис.1. Загальна схема

Причина, по якій для кожного жанру музики розроблена та навчена окрема нейромережа полягає в тому, що нейромережі типу GAN є дуже складними у навчанні та розробці, оскільки потребують великих потужностей та довго навчаються. Розробка GAN нейромережі для генерації всіх жанрів музики, яка навчалася на всіх жанрах музики значно ускладнює архітектуру, потребує більше потужностей та часу на навчання.

Класифікація тексту є частиною NLP (Natural Language Processing) області, яку вважають областю штучного інтелекту через її

складність. Для задачі класифікації тексту [4] було обрано класифікатор на основі архітектури двонаправленої рекурентної нейронної мережі (BRNN) з “ефектом пам’яті” – LSTM, оскільки рекурентні нейромережі створені для роботи з послідовністю, двонаправленість забезпечує обробку слів в прямому та зворотньому напрямках, що збільшує якість передбачень, а Long short-term memory (LSTM) дозволяє розуміти зміст більш довгих послідовностей. Класифікатор навчався на наборі даних, розділених на текстові дані (твіти користувачів мережі Twitter) [5], які належать одному з шести класів емоцій.

Перед початком тренування класифікатора, дані були очищені від повторюваних та стоп-слів розділені на тренувальний (70%) та валідаційний (30%) набори, токеновані за допомогою токенайзера та пропущені через Glove Embedding. Щоб досягти точності у 96% класифікатору необхідно близько 10 хвилин. Дані для перевірки точності класифікації були обрані з книги “Гобіт, або Туди і Звідти”, автора Дж.Р.Р. Толкіна, оскільки вони є складнішими за інтерпретацією, ніж тренувальні дані, результат показано на рис.2 (Повний текст: *“Not a nasty, dirty, wet hole, filled with the ends of worms and an oozy smell, nor yet a dry, bare, sandy hole with nothing in it to sit down on or to eat: it was a hobbit-hole,*

```
Not a nasty, dirty, wet hole, filled with the ends of worms and an oozy smell
1/1 [=====] - 1s 604ms/step
1/1 [=====] - 1s 844ms/step
joy : 0.9873606562614441
```

and that means comfort.”)

Рис.2. Результат класифікації

Друга частина є складнішою у побудові та навчанні. За основу взятий підхід, запропонований Google AI на конференції ICLR 2019 [6]. Він полягає в тому, щоб вчити нейромережу з GAN архітектурою генерувати не саму музичну послідовність, а музичні спектрограми, оскільки GAN-и показують найкращий результат на задачах генерації зображень. Архітектура GAN складається з генератора, чия задача генерувати максимально якісні зображення (fake samples) та дискримінатора, чия задача повертати до генератора відсоток “вдалих”

зображень – це зображення, які дискримінатор класифікував як реальні (real samples), хоча вони згенеровані генератором. Завдяки такій архітектурі генератор та дискримінатор “навчають” один одного. Для цього проєкту генератор та дискримінатор нейромережі мають згорткову архітектуру, оскільки математична операція згортки застосовується для роботи з зображеннями. На відміну від інших нейромереж, де задачею є мінімізація функції втрат, задачею GAN нейромережі є стабілізація функції втрат та максимізація функції точності. Досягнення такої стабільності є найскладнішою частиною розробки нейромереж такого типу, оскільки стабілізація функцій є складним математичним процесом. На основі публікації [7] вдалося частково стабілізувати GAN нейромережу (рис.3).

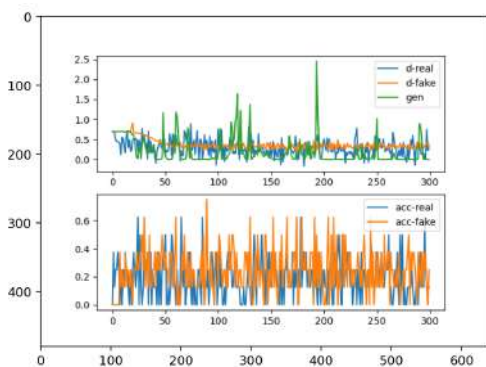


Рис 3. Графіки функцій втрат та точності

Наведена нейромережа тренувалася на спектрограмах, створених з 30-секундних композицій жанру блюз. Для забезпечення стабільності було застосовано кілька підходів: розділене навчання дискримінатора: спочатку на даних реальних спектрограмах, після навчання на згенерованих спектрограмах, заміна міток класів реальних та згенерованих зображеннях: замість міток 0 (згенеровані зображення) та 1 (реальні зображення) використано функцію генерації рандомних чисел Гаусівського розподілу – межі генерації для згенерованих зображень: 0 - 0.2, для реальних зображень: 0.7 - 1.1., заміна функцій Sparse градієнту (MaxPool, ReLu) на Conv2dTranspose та LeakyReLu відповідно.

Результат навчання на 50 епохах (1.5 години, 10 батчів по 10 об'єктів) наведено на рис.4.

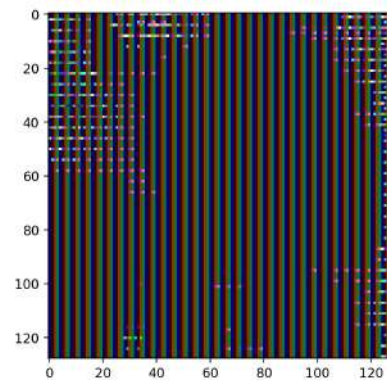


Рис.4. Згенерована спектрограм після навчання протягом 50 епох

Без застосування цих підходів нейромережа замість спектрограм буде генерувати шум (рис.5), функція втрат у дискримінатора після 2 епохи буде близька до 0, тоді як функція втрат генератора буде постійно зростати (рис.6.)

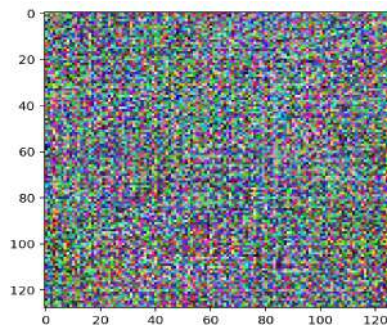


Рис 5. Згенероване шумове зображення.

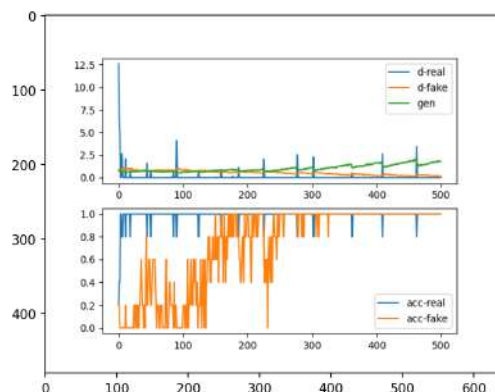


Рис.6. Графіки функцій втрат та точності

Висновки. Розроблено програмне забезпечення на основі архітектур нейронних мереж для генерування музики з врахування емоційної складової та розглянуто підходи для оптимізацій та покращення якості генеративних змагальних моделей нейронних мереж .

Список інформаційних джерел

1. Generative Adversarial Networks / Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio // Advances in Neural Information Processing Systems 27 (NIPS 2014). – 2014. – Vol. 27. – P. 2672–2680
2. Nature Inspired Computation and Swarm Intelligence / Xin-SheYang // Academic Press. – 2020. – 442 p.
3. Music Emotion Recognition: Toward new, robust standards in personalized and context-sensitive application / J.S.Gómez-Cañón; E.Cano,T.Eerola; P.Herrera; Xiao Hu; Yi-Hsuan Yang // IEEE Signal Processing Magazine, Vol. 38 (6), 2021, P. 106-114
4. Review of Text Classification in Deep Learning / Qi Wang, Wenling Li, Zhezhi Jin // Open Access Library Journal. – Vol. 8 (03). – 2021. – P. 1-8. 5. Emotions dataset for NLP. URL:<https://www.kaggle.com/datasets/praveengovi/emotions-dataset-for-nlp>
6. GANSYNTH: ADVERSARIAL NEURAL AUDIO SYNTHESIS / J.Engel, K.K.Agrawal, S.Chen, I.Gulrajani, C.Donahue, A.Roberts // ICLR. – 2019. URL: <https://arxiv.org/abs/1902.08710>
7. How to Train a GAN? Tips and tricks to make GANs work / Soumith Chintala, Emily Denton, Martin Arjovsky, Michael Mathieu // NIPS 2016. URL: <https://github.com/soumith/ganhacks>