

# Компонирање македонска песна

## Генерирање на текст и музика за песна со помош на машинско учење

Ангела Најдоска, Емилија Котевска, Тамара Маркачевиќ  
Факултет за електротехника и информациски технологии- КТИ и КХИЕ  
Скопје, Македонија  
angela.najdoska17@gmail.com, emilijakotevska@gmail.com, markacevic@gmail.com

**Анстракт**— Овој труд дава преглед на тоа колку е предизвикувачко создавањето песна и дава увид во тоа како развиеме генератор на текст и музика за песна на македонски јазик. Направивме истражување во различни области како што се вештачка интелигенција и обработка на природен јазик за да може да ги совладаме различните техники за обработка и генерирање на текст и да можеме да ги користиме на наш начин. Создаваме и тренираме наш модел кој ќе го изгенерира текстот. За создавање на музиката откривме веќе истрениран модел од библиотеката Magenta од Tensorflow кој автоматски генерира секвенца од ноти. Со помош на овој готов модел генерираме секвенци кои подоцна ги предаваме на програма Interpolate од Magenta Studio, која интерполира помеѓу две секвенци од ноти и на тој начин го заокружува процесот на креирање музика.

**Клучни зборови**—Python, генерирање текст; македонска песна; Keras; музика; модел; длабоко учење; машинско учење; Magenta; Tensorflow.

### I. ВОВЕД

Машинското учење, како важен сегмент од вештачката интелигенција е се поприсутен во нашето секојдневие. Генерирањето на текст со помош на машинско учење и длабоко учење е популарно во секоја индустрија, особено кај мобилните телефони, апликации, и data-science. Дури и новинарството користи генерирање текст за да го олесни процесот на пишување. Секојдневно се среќаваме со генерирање на текст: автоматското дополнување на iMessage, пребарување на Google, паметното пишување на Gmail се само неколку примери. Користејќи автоматско генерирање на текст можеме да олесниме многу процеси, како на пример композирањето на песна. Толку затоа, во овој проект создаваме и тренираме модел за генерирање на текст, а покрај генерирањето на текст, применуваме Длабоко учење за да создадеме и музика.

### II. ДАТАБАЗА ЗА ТЕКСТОТ

За генерирање на текстот на песната, креираме наша база која се состои од текстовите од сите песни на нашиот пеач Тоше Проески. Имаме 64 песни, секоја со различен број на стихови, и различна метрика. Вкупниот број на зборови во базата изнесува 10.731.

### III. ПРЕТПРОЦЕСИРАЊЕ НА ПОДАТОЦИТЕ

Пред да му ги дадеме податоците за тренирање на моделот, неопходно е да направиме нивно претпроцесирање, што подразбира креирање на корпус врз кој моделот ќе се тренира, негова токенизација и изедначување на бројот на токени во секој стих.

Најпрво ја прочистуваме базата со песните, ослободувајќи се од интерпунктските знаци и претварајќи ги сите букви во мали. Потоа го формираме корпусот така што секој стих од базата претставува еден негов елемент. Со користење на Tokenizer од библиотеката keras ги токенизираме податоците претварајќи ги во секвенци од токени со различна должина, а потоа сите секвенци ги доведуваме до иста должина преку pad\_sequences од keras библиотеката. Секоја од секвенците ја делиме на predictors (features) и label, притоа користејќи one-hot encoding за излезот.

Првата секвенца поделена на:  
predictors: [ 0 0 0 0 0 0 0 0 0 0 0 0 460]  
label: 14

Слика 1 : Корпусот поделен на секвенци

#### IV. МОДЕЛ

Моделот кој го користиме за да го изгенерираме текстот на нашата песна е секвенцијален тип на модел кој вклучува Рекурентна невронска мрежа. Тој се состои од:

- “Embedding Layer” со број на влезови еднаков на бројот на различни зборови во корпусот, 5 излези и input\_length еднакво на бројот на токени во секвенца -1.
- 2 сокриени слоја од “Gated Recurrent Unit (GRU)” рекурентна невронска мрежа, по што следува Dropout од 20%. Бројот на неврони во двата слоеви е растечки така што првиот слој содржи 80, а вториот 120 неврони.
- Излезен (“Dense”) слој со број на излези еднаков на бројот на различни зборови во корпусот и softmax активациона функција.

Функција на цена за нашиот модел ќе биде “categorical\_crossentropy”, неа ја оптимизираме со “adam optimizer” и притоа за следење на точноста која ни ја дава моделот при негово тренирање користиме метрика за точност- “accuracy”.

#### V. ТРЕНИРАЊЕ НА МОДЕЛОТ И ГЕНЕРИРАЊЕ ТЕКСТ

Моделот го тренираме во 400 епохи, а притоа користиме и “EarlyStop callback” со цел да спречиме негово претренирање[1]. Од веќе постоечкиот корпус правиме нов корпус чии елементи се зборови, со цел од него да избереме рандом збор кој ќе послужи како почетен збор врз кој ќе се изгенерира целата песна.

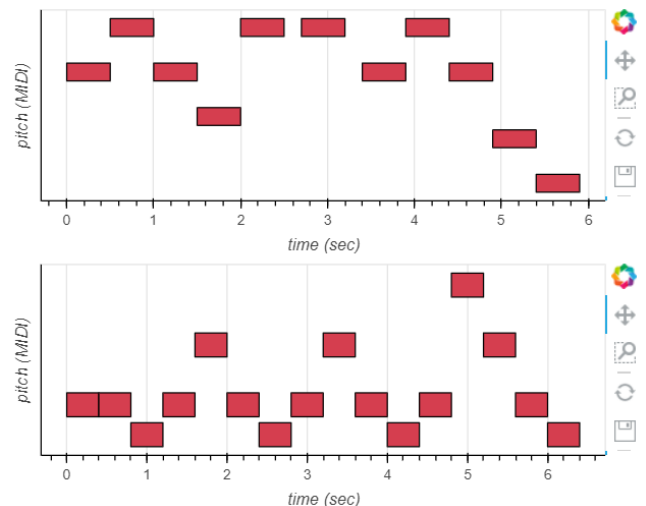
Песната ја генерираме на тој начин што користиме функција која генерира еден стих. Притоа првиот изгенериран збор го користи принципот one-to-one, додека сите наредни зборови од стихот се генерираат како many-to-one[2]. Почетен збор за секој нов стих е последниот збор од претходниот. Произволно избираме песната да се состои од 4 строфи со по 4 стиха, а секој стих да содржи 6 зборови.

#### VI. MAGENTA

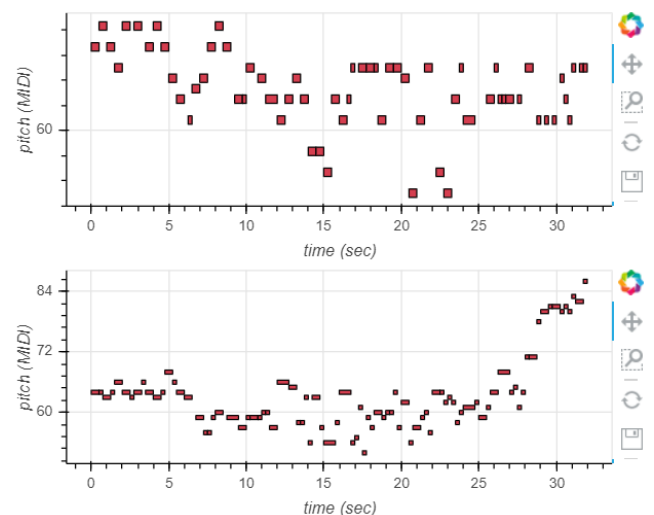
За генерирање на музика ја користиме Python библиотеката Magenta дистрибуирана од Tensorflow. Покрај Magenta особено важна ни е и библиотеката node\_seq која овозможува апстрактна репрезентација на серија од ноти базирана на MIDI стандардот (Musical Instruments Digital Interface) кој една серија од ноти ја претставува со нивната висина на тон, инструмент, брзина на удар, времетраење и растојание меѓу нотите [3].

#### VII. MELODY RNN И ГЕНЕРИРАЊЕ МУЗИКА

Библиотеката Magenta нуди веќе истренирани модели кои можат да се искористат. Еден од нив е и моделот Melody RNN- модел на Рекурентна Невронска мрежа за генерирање секвенца од ноти врз база на почетно зададена секвенца[4]. Ние го користиме овој модел за да изгенерираме две различни музички секвенци. Најпрво со помош на функцијата NodeSequences од библиотеката node\_seq ги креираме почетните секвенци кои всушност се дел од две песни од Тоше Проески „Ледена“ и „По тебе“. Потоа на Melody RNN моделот му задаваме одредени параметри како што се температура(степен на случајност на генерирање нота), број на чекори(од него зависи должината на изгенерираната секвенца), темпо. По генерирање на секвенците нив ги зачувуваме во midi формат.



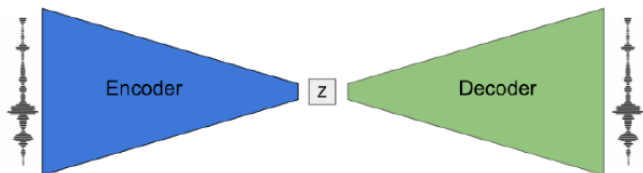
Слика 2: Почетни секвенци-делови од песни на Тоше Проески



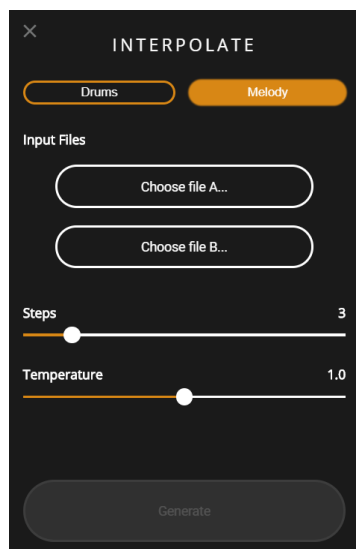
Слика 3: Секвенци изгенерирани со Melody RNN

## VIII. MAGENTA STUDIO

Magenta Studio е колекција на самостојни апликации изградени врз алатки и модели од Magenta[5]. Оваа колекција ги содржи апликациите: Continue, Groove, Generate, Drumify и Interpolate кои овозможуваат примена на моделите од библиотеката Magenta врз midi датотеки[6]. Interpolate е апликација која зема два удари во тапан или две мелодии како влезови. Може да креира најмногу 16 датотеки кои ги комбинираат квалитетите на двете влезни датотеки. Корисна е за спојување музички идеи или за создавање непречено софпаѓање меѓу нив[7]. Ние ја користиме токму неа за да ги споиме двете претходно изгенерирани музички секвенци. Оваа апликација се базира на моделот Music VAE кој има структура на варијациски авто-кодер. Еден начин да се размислува за VAE е како мапирање од MIDI во компресиран простор во кој слични музички обрасци се собираат заедно. Секој ваш влезен образец е претставен со позиција на оваа мапа. Interpolate повлекува линија помеѓу овие позиции и враќа клипови по оваа линија. Бројот на вратени датотеки е поставен со лизгачот steps. Секоја од апликациите има лизгач за температура. Температурата е параметар што се користи за земање примероци во последниот слој на невронската мрежа. Овој параметар служи за генерирање на случајноста. Повисоките вредности создаваат поголема варијација, а понекогаш дури и хаос, додека пониските вредности се поконзервативни во нивните предвидувања[8].



Слика 4: Структура-дел од моделот Music VAE



Слика 5: Interpolate апликација

## IX. РЕЗУЛТАТИ

На погоре опишаниот начин генерираме неколку текстови за нашата песна, по што се одлучуваме за оној кој нам најмногу ни се допаѓа.

Конечно, музиката за нашата песна ја добиваме после примена на интерполација на двете претходно изгенерирани секвенци, со помош на Interpolate апликацијата од Magenta Studio. Ги прикачуваме двете мелодиски секвенци, одбираме температура еднаква на 1.5 и генерираме 5 датотеки, со подесување на параметарот steps. Потоа овие датотеки со помош на online конкатенатор на midi датотеки ги спојуваме во една, со цел подолга музичка датотека[9]. Добивме музика со времетраење од 32 секунди.

во окото сјајот не поминува не  
не е месечината во ноќва што  
ме привлекува што во глава ти  
фали си слатка но и студена

ми илјада и двеста прегратки насмевка  
и ледена и медена и да  
ми илјада и двеста прегратки насмевка  
и ледена и медена и да

усни на усни да те води  
познат и двеста дена суви со  
чуда по тебе изгорев јас и  
каде ли траг на вратот криеш

дали е доцна нешто да сменам  
да те најдам ради нас патуваш  
пак не е грубо ако е  
од тебе во него на ум

Слика 6: Текстот за песната

## X. АНАЛИЗА НА РЕЗУЛТАТИ

Обидувајќи се да го креираме најдобриот можен модел за генерирање на текстот, поминавме низ долг процес на обиди каде пробувавме различен тип на невронски мрежи, различен број на скриени слоеви, број на неврони, активациски функции.

Најпрво се обидовме со Long Short Term Memory невронска мрежа (LSTM), која се состоеше од “Embedding Layer” со број на влезови еднаков на бројот на различни зборови во корпусот, 7 излези и input\_length еднакво на бројот на токени во секвенца -1; Два скриени слоја со Dropout од 20% после секој од нив, и опаѓачки број на неврони во слоевите и излезен (“Dense”) слој со број на излези еднаков на бројот на различни зборови во корпусот и softmax активациска функција. Со ваквиот модел добивме точност од 65%.

Употребата на “Gated Recurrent Unit (GRU)” рекурентна невронска мрежа, ја зголеми точноста, споредено со LSTM. Исто така заклучивме дека со зголемување на број на скриени слоеви, кај двата типа на невронски мрежи доведува до намалување на точноста.

Се обидовме и со примена на gelu активациска функција наместо softmax, меѓутоа резултатите беа катастрофални, добивме песна која се состоеше од по еден збор во секој стих, при што сите стихови од една строфа беа идентични.

Со намалување на бројот на излези од “Embedding” слојот, се подобруваше точноста па од 7 излези се симнавме на 5. Повисока точност добивме кога го заменивме опаѓачкиот број на неврони во двата скриени слоеви со растечки.

Конечно, со отстранување на Dropout-от после првиот скриен слој постигнавме точност од 72%.

## XI. ЗАКЛУЧОК

Музиката е меѓу ретките уметности што се пренесува од една на друга генерација. Несомнено е дека нејзиното место е многу важно. Затоа целта на овој проект беше да креираме добар модел за генерирање на текст за песна, а воедно и музика за истата користејќи ги сите песни кои се на македонски јазик од Тоше Проески. Во иднина крајниот резултат би можел уште повеќе да се подобри со додавање на повеќе песни од други македонски изведувачи во базата и со генерирање на звуци од друг инструмент, пример тапани.

## КОРИСТЕНА ЛИТЕРАТУРА

[1][https://www.tensorflow.org/api\\_docs/python/tf/keras/callbacks/EarlyStopping](https://www.tensorflow.org/api_docs/python/tf/keras/callbacks/EarlyStopping)

[2] <https://stackabuse.com/solving-sequence-problems-with-lstm-in-keras/>

[3] <https://magenta.tensorflow.org/>

[4] Magenta, “magenta/magenta,” GitHub, 07-Apr-2020. [Online]. Available: <https://github.com/magenta/magenta>

[5] <https://magenta.tensorflow.org/studio/standalone>

[6] <https://magenta.tensorflow.org/studio>

[7]<https://magenta.tensorflow.org/studio/standalone#interpolate>

[8] <https://magenta.tensorflow.org/studio/standalone#usage>

[9] <https://www.ofoct.com/merge-midi-files>

[10] “MusicVAE: Creating a palette for musical scores with machine learning.,” Magenta, 15-Mar-2018. [Online]. Available: <https://magenta.tensorflow.org/music-vae>