

Perancangan Sistem Generate Lirik Lagu Berbasis Website dengan Menggunakan Algoritma Markov Chain

Agus Yanto¹, Eric², Jerry³, Vendryan⁴

^{1,2,3,4} Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Internasional Batam
Email: khujerry02@gmail.com

ABSTRAK

Musik merupakan suatu seni yang sering didengar baik sedang sedih maupun bahagia dari berbagai kalangan. Lirik dari sebuah lagu ini sangat sulit dibuat jika belum memiliki pengalaman di dalam dunia permusikan. Dengan adanya teknologi kecerdasan buatan maka lirik dari sebuah lagu ini bisa dibuatkan agar membantu para masyarakat ataupun artis yang baru terjun ke dunia musik. Dengan menggunakan Natural Language Processing agar mesin bisa memahami bahasa manusia dan juga menggunakan algoritma markov chain untuk menghitung probabilitas dari kata sebelumnya untuk menentukan kata berikutnya maka akan bisa dibuatkan sebuah sistem kecerdasan buatan yang bisa melakukan generate lagu. Hasil dari penelitian ini adalah sebuah website yang bisa melakukan generate lirik lagu baru dari artis-artis yang ada yang diberi nama Lyrichisty.

Kata kunci: Lirik Lagu, NLP, Markov Chain.

ABSTRACT

Music is an art that is often heard whether you are sad or happy from various circles. The lyrics of a song are very difficult to make if you don't have experience in the world of music. With the existence of artificial intelligence technology, the lyrics of a song can be made to help the public or artists who are just getting into the world of music. By using Natural Language Processing so that machines can understand human language and also using the markov chain algorithm to calculate the probability of the previous word to determine the next word, an artificial intelligence system can be created that can generate songs. The result of this research is a website that can generate new song lyrics from existing artists named Lyrichisty.

Keywords: Song Lyrics, NLP, Markov Chain.

1. PENDAHULUAN

Musik adalah suatu seni yang sering didengarkan dari semua kalangan, baik itu anak-anak, remaja, bahkan orang tua. Selain memiliki fungsi sebagai hiburan, musik juga berfungsi sebagai sarana penyampaian komunikasi [1]. Para musisi-musisi sering mengeluarkan album musik mereka dimana album musik tersebut berisikan lagu-lagu yang telah mereka buat. Di Dalam lagu tersebut berisikan pesan yang tersirat seperti jika lagunya sedih berarti pesan yang ingin disampaikan ke dalam lagu itu pastinya akan sedih. Pesan yang disampaikan ke dalam sebuah lagu tersebut akan bisa dimengerti jika kita memahami lirik dari lagu tersebut. Lirik-lirik yang ada di dalam lagu tersebut adalah sebuah karangan dari sang musisi yang melalui banyak pengalaman dan perubahan.

Dengan berkembangnya teknologi yang ada dunia, segala sesuatu pekerjaan yang ada dunia sekarang lebih dimudahkan. Teknologi juga sudah masuk kedalam lingkup musik, dimana teknologi ini akan sangat membantu para musisi. Seperti teknologi penghilang noise, auto-tune, dan sebagainya. Bagi para musisi-musisi yang baru terjun dan yang baru akan terjun ke dunia musik, pasti akan kesulitan membuat sebuah lirik lagu. Karena sebuah lirik lagu akan membutuhkan pengalaman dan pemahaman yang cukup [2]. Dengan adanya teknologi ini juga seharusnya bisa membantu para musisi dalam menuliskan lirik lagu. Dengan menggunakan teknologi Artificial Intelligence (Kecerdasan Buatan) maka seorang musisi akan bisa membuat sebuah lirik lagu.

Artificial Intelligence adalah sebuah rancangan oleh manusia dengan memanfaatkan kemajuan teknologi sehingga dapat melakukan sebuah tugas ataupun pekerjaan tertentu sesuai dengan apa yang diprogramkan [3]. Untuk membuat sebuah teknologi yang bisa membuat sebuah lirik lagu maka bisa menggunakan salah satu teknologi yang ada di Artificial Intelligence yaitu Natural Language Processing (NLP). Dengan NLP maka mesin akan bisa mengetahui bahasa manusia lebih akurat.

Pada NLP nanti akan digunakan juga sebuah algoritma yaitu Markov Chain. Dimana algoritma Markov Chain ini akan menghitung probabilitas kemungkinan kata yang akan dikeluarkan untuk baris berikutnya. Dengan menerapkan semua teknologi dan algoritma tersebut maka akan bisa membuat sebuah teknologi yang bisa membuat sebuah lirik lagu.

Pada penelitian ini, penulis akan membuat sebuah Artificial Intelligence yang bisa membuat sebuah lirik lagu yang bergenre cinta. Pada sistem ini akan menggunakan NLP dan juga algoritma Markov Chain yang kemudian akan di implementasikan ke dalam website dengan menggunakan Flask dan menggunakan website PythonAnywhere. Yang dimana website tersebut akan bisa digunakan oleh musisi-musisi.

2. TINJAUAN PUSTAKA

Pada penelitian Rosyadi berfokus kepada rancang bangun sebuah chatbot yang bisa digunakan untuk mengetahui lowongan pekerja di aplikasi WhatsApp. Tujuan dari penelitian ini adalah untuk mempermudah para pekerja yang mencari lowongan dengan hanya menggunakan aplikasi WhatsApp. Metode yang digunakan dalam penelitian ini adalah metode waterfall untuk membuat website yang berguna untuk memasukan informasi lowongan pekerjaan dan juga dibantu dengan metode Natural Language Processing untuk melakukan pembuatan sistem chatbot. Hasil dari penelitian ini adalah sebuah Chatbot yang bisa mencari lowongan pekerjaan di aplikasi WhatsApp dan juga sebuah website yang dimana kegunaannya untuk mengatur

kebutuhan yang berhubungan dengan lowongan pekerjaan yang akan dimasukkan kedalam chatbot [4].

Pada penelitian Riedel ini berfokus kepada penyelidikan terhadap pengaruh Coronavirus terhadap tangga lagu tahun 2020. Tujuan dari penelitian ini adalah untuk mengetahui seberapa tinggi pengaruh coronavirus untuk merubah genre terhadap lagu. Metode yang digunakan dalam penelitian ini adalah metode Natural Language Processing yang berfungsi untuk melakukan lirik mining dan kemudian akan melihat tangga lagu sebelum masuknya coronavirus dan sesudah masuknya coronavirus. Hasil dari penelitian ini adalah adalah beberapa genre yang sedang naik seketika langsung turun dan digantikan oleh genre yang lain [5].

Selanjutnya penelitian yang dilakukan oleh Mustakim berfokus kepada memprediksi kejadian bencana alam di Provinsi Riau. Tujuan dari penelitian ini adalah untuk mengetahui bencana apa yang akan terjadi di Riau di kedepan harinya. Metode yang digunakan dalam penelitian ini adalah dengan cara mengumpulkan data, pengolahan data, dan kemudian melakukan implementasi algoritma markov chain untuk bisa melakukan prediksi bencana alam sesuai dengan data bencana yang telah terjadi sebelumnya. Hasil dari penelitian ini adalah banyak dari bencana alam akan naik dan turun karena dipengaruhi dari segi geografis Provinsi Riau, contohnya seperti bencana banjir diprediksi akan mengalami penurunan 2% di tahun 2019 dan bencana puting beliung akan mengalami kenaikan sebanyak 4% pada tahun 2019 di Riau [6].

Dan penelitian yang dilakukan oleh Hidayati ini hampir sama dengan penelitian yang dilakukan Mustakim karena pada penelitian ini topiknya adalah memprediksi bencana alam yang akan terjadi di Kota Semarang. Dan hasilnya adalah bencana alam banjir akan mengalami penurunan sebanyak 13% di tahun 2021 dari tahun 2020 dan juga ada yang mengalami kenaikan seperti kebakaran yang akan mengalami peningkatan 3% dari tahun 2020 [7].

Tujuan dari penelitian ini adalah untuk membuat sebuah kecerdasan buatan yang akan membuat sebuah lirik lagu berdasarkan artis berbasis website. Metode yang digunakan didalam penelitian ini adalah dengan menggunakan NLP seperti pada penelitian Rosyadi dan penelitian Riedel yang berguna untuk membuat sebuah mesin atau sistem tersebut paham akan Bahasa manusia, dan kemudian mengimplementasikan sebuah algoritma Markov Chain seperti yang dilakukan pada penelitian Mustakim dan juga penelitian Hidayati yang bisa menghitung probabilitas yang akan terjadi berikutnya sehingga lirik yang akan terbentuk dari kecerdasan buatan ini akan sangat padu dan kemudian akan dimasukkan kedalam sebuah website dengan menggunakan Flask.

2.1. Natural Language Processing

Natural Language Processing (NLP) adalah cabang dari Artificial Intelligence (Kecerdasan Buatan) yang berhubungan dengan interaksi antara komputer dengan manusia menggunakan bahasa manusia sehari-hari [8]. NLP menggabungkan linguistic komputasi – pemodelan bahasa manusia berbasis aturan dengan model statistik, machine learning (pembelajaran mesin) dan juga deep learning (pembelajaran mendalam). Bersama dengan semua teknologi diatas maka itu akan memungkinkan sebuah program komputer untuk memproses bahasa manusia dalam bentuk teks maupun dalam bentuk suara dan memahami maksud sebenarnya dan lengkap dengan sentimen yang diucapkan atau dituliskan [9].

Ada dua fase utama pada NLP, yaitu data preprocessing dan juga pengembangan algoritma [10]. Data preprocessing adalah persiapan yang akan dilakukan oleh peneliti untuk

membersihkan dan melakukan persiapan kepada data yang akan digunakan sehingga data tersebut bisa dikerjakan oleh suatu algoritma yang akan dipakai. Ada beberapa Teknik dalam tahap preprocessing yaitu:

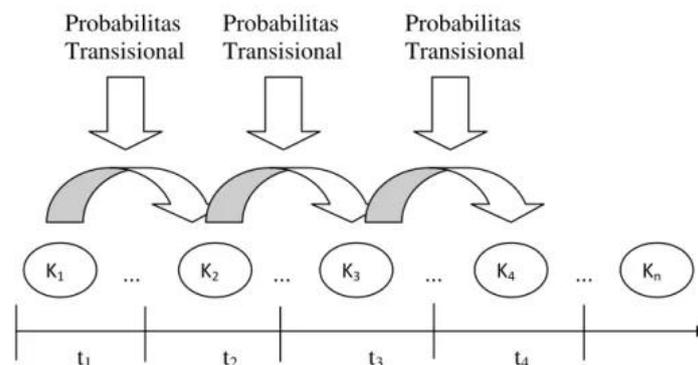
- Tokenization
- Stopword removal
- Lemmatization dan stemming
- Part of speech tagging

Setelah melakukan tahap data preprocessing selanjutnya akan masuk kedalam tahap pengembangan algoritma dimana di tahap ini algoritma dikembangkan untuk memproses data pada tahap berikutnya. Ada banyak jenis algoritma yang bisa dipakai, akan tetapi ada 2 algoritma utama yang sering dipakai antara lain:

- Algoritma berbasis aturan
- Algoritma berbasis machine learning

2.2. Markov Chain

Markov Chain atau biasa disebut juga dengan rantai markov adalah suatu teknik riset yang menggunakan gejala Brownian atau gerakan lurus secara terus-menerus tetapi tidak beraturan untuk segala jenis sistem dan proses bisnis dalam menjelaskan perubahan yang akan terjadi di masa yang akan datang [6]. Model Rantai Markov ditemukan oleh seorang ahli dari Rusia yang bernama A.A. Markov pada tahun 1906, yaitu: "Untuk setiap waktu t , ketika kejadian adalah K_t dan seluruh kejadian sebelumnya adalah $K_t(j), \dots, K_t(j-n)$ yang terjadi dari proses yang diketahui, probabilitas seluruh kejadian yang akan datang $K_t(j)$ hanya bergantung pada kejadian $K_t(j-1)$ dan tidak bergantung pada kejadian-kejadian sebelumnya yaitu $K_t(j-2), K_t(j-3), \dots, K_t(j-n)$." Peristiwa rantai Markov ini dijelaskan dalam "Gambar 1".



Gambar 1. Peristiwa Rantai Markov

Jika dilihat dari gambar diatas kejadian-kejadian ini bersifat berantai, oleh karena itu teori ini disebut dengan rantai Markov. Rantai Markov akan menjelaskan pergerakan beberapa variabel dalam suatu periode waktu yang akan datang berdasarkan pergerakan variabel masa kini [7]. Secara sistematis dapat juga ditulis seperti pada persamaan (1) dibawah ini.

$$Kt(j) = P \times Kt(j - 1) \tag{1}$$

Dimana:

$Kt(j)$ = Peluang kejadian pada $t(j)$

P = Probabilitas transional

$t(j)$ = Waktu ke -j

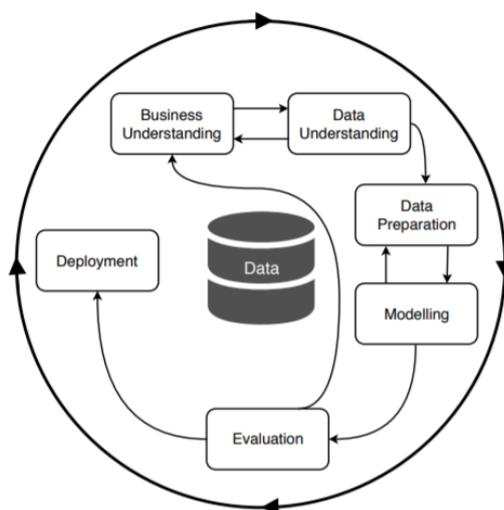
Peluang kejadian $Kt(j)$ dinyatakan ke dalam bentuk vektor sehingga jumlah seluruh selnya akan selalu 100%

2.3. Flask

Flask merupakan web framework yang ditulis dalam bahasa Python. Flask memiliki dua external libraries yaitu WSGI toolkit dan Jinja2 template engine. Flask merupakan jenis micro framework yang tidak memerlukan library tertentu dalam penggunaannya. Flask dapat menggunakan ekstensi untuk menambahkan fitur dan komponen yang sudah disediakan oleh pihak ketiga dan tidak terpasang secara standar pada Flask seperti Form Validation, Upload Handling, dan Database [11].

3. METODE PENELITIAN

Pada penelitian kali ini, tim penulis akan menggunakan metode CRISP-DM seperti yang ditunjukkan pada "Gambar 2" untuk melakukan pengembangan terhadap pembelajaran mesin yang nantinya akan bisa membuat sebuah lirik lagu baru berdasarkan artis.



Gambar 2. Tahapan Metode CRISP-DM

Pada tahapan CRISP-DM ada 6 tahapan yaitu Business Understanding, Data Understanding, Data Preparation, Modelling, Evaluation, dan yang terakhir adalah Deployment. Setiap arah panah berfungsi untuk menunjukkan siklus dan menggambarkan bahwa data mining dapat menghasilkan sebuah solusi, pengetahuan, dan pemahaman bisnis.

1. Business Understanding
Pada tahap ini akan dilakukan terlebih dahulu pengamatan terhadap lirik lagu. Bagaimana

sebuah lirik lagu tersebut bisa terbentuk dan juga bagaimana keserasian dari setiap kalimat dari lirik lagu tersebut.

2. **Data Understanding**
 Pada tahap data understanding ini, tim penulis akan mulai mengumpulkan data-data lirik lagu yang nantinya akan digunakan sebagai dataset untuk pembuatan kecerdasan buatan ini.
3. **Data Preparation**
 Di sini, akan dilakukan pembersihan data yang memiliki data kosong, data berganda (double) dan juga data yang memiliki kalimat-kalimat pengulangan. Dan data ini akan digunakan untuk data training.
4. **Modelling**
 Di tahap modelling, ini akan dilakukan implementasi NLP dengan menggunakan algoritma markov chain. Data akan di train dengan markov chain sehingga nanti akan muncul lirik lagu yang baru.
5. **Evaluation**
 Pada tahap evaluation ini, tim penulis akan bertanya kepada orang-orang apakah lirik lagu yang di-generate dari sistem sudah dibuat tim penulis sesuai atau tidak, jika tidak maka akan dilakukan training ulang.
6. **Deployment**
 Setelah semua tahapan diatas selesai, maka tim penulis akan melakukan deployment ke dalam bentuk website dimana dengan menggunakan framework flask.

4. HASIL DAN PEMBAHASAN

4.1. Business Understanding dan Data Understanding

Pada tahap ini tim penulis bisa mengetahui ternyata setiap penyanyi memiliki gaya penulisan yang berbeda-beda terhadap lirik lagu. Maka dari itu nanti penulis akan mengelompokkan lirik-lirik itu berdasarkan dengan penyanyi-penyanyinya tersendiri. Berikut adalah data yang telah tim penulis dapatkan pada "Gambar 3".

| | A | B | C | D | E |
|-----|-----------------------------------|-------------------|---|---|---|
| 707 | Sampai Akhir Waktu | Yovie & Nuno | Ku akan datang lagi | | |
| 708 | Sakit Hati | Yovie & Nuno | Ku tahu engkau pasti tahu | | |
| 709 | Malam Mingguku | Yovie & Nuno | Lihat langit biru | | |
| 710 | Demi Hati | Yovie & Nuno | Hari demi hari dan juga malam | | |
| 711 | Lebih Baik Aku | Yovie & Nuno | Adakah benar ku terlambat | | |
| 712 | Seperti Bintang | Yovie & Nuno | Andai saja engkau tahu | | |
| 713 | Adu rayu | Yovie Tulus Glenn | Maukah lagi kau mengulang ragu | | |
| 714 | Berawal dari tatap | Yura Yunita | Berawal dari tatap | | |
| 715 | Peri Cintaku | Ziva Magnolya | Di dalam hati ini, hanya satu nama | | |
| 716 | Gimme Love | Joji | Gimme-gimme love, gimme-gimme love | | |
| 717 | MODUS | Joji | Speed it up, slow it down | | |
| 718 | Glimpse of Us | Joji | She'd take the world off my shoulders if it was ever hard to move | | |
| 719 | Die For You | Joji | Swear I couldn't sleep a wink last night | | |
| 720 | Sanctuary | Joji | Go ahead and park after dark | | |
| 721 | Run | Joji | I fell for your magic, I tasted your skin | | |
| 722 | TEST DRIVE | Joji | She don't wanna tell lies | | |
| 723 | SLOW DANCING IN THE DARK | Joji | I don't want a friend | | |
| 724 | WANTED U | Joji | Here I am | | |
| 725 | WHY AM I STILL IN LA (feat. Shloh | Joji | Girl, you choke me sitting airtight (airtight) | | |
| 726 | WANTED U | Joji | Yeah, all my friends no fun | | |

Gambar 3. Dataset Lirik Lagu

Dari dataset diatas dapat disimpulkan bahwa:

1. Data lirik lagu diatas berjumlah 726 data, yang dimana data tersebut tim penulis kumpulkan sendiri dengan mencari judul lagu yang termasuk kedalam topik cinta berdasarkan artis-artis.
2. Ada 3 atribut di dalam data tersebut antara lain adalah Judul_lagu, Artis dan juga Lirik_lagu.

4.2. Data Preparation

Pada tahapan ini, tim penulis mulai melakukan data cleaning dengan menghilangkan data lirik yang mempunyai baris yang kosong dan juga yang mempunyai empty string seperti yang ditunjukkan pada "Gambar 4". Tidak hanya itu saja tim penulis juga menambahkan banyak persiapan terlebih dulu sebelum memasuki tahap modelling

```
# CLEANING DATA
def cleaning_data(df_lyric):
    """Melakukan cleaning pada dataset lirik cinta
    Menghilangkan baris yang ada null dan empty string
    """
    nan_value = float('NaN')
    df_lyric.replace('', nan_value, inplace=True)
    df_lyric.dropna(subset=['lyric'], inplace=True)
    df_lyric.head()

    return df_lyric
```

Gambar 4. Menghapus Null pada Baris Lirik

Adapun beberapa tahap persiapan tersebut adalah dengan cara mengubah seluruh lirik lagu menjadi huruf kecil semua, mulai menambahkan BEGIN dan END untuk menandakan permulaan dan akhir pada lirik. Pada data terdapat jenis newline yang berbeda yaitu \r\n dan \n sehingga harus diubah menjadi satu dan dikasih penanda dengan menggunakan NEWLINE seperti yang ditunjukkan pada "Gambar 5". Dan juga mengubah koma dengan menggunakan COMMA. Kemudian juga mulai melakukan seragaman pada lirik lagu, misalnya "s'lalu" menjadi "selalu" agar lebih konsisten di dalam pemodelan nantinya. Dan selanjutnya akan menghapus lirik yang sama seperti pengulangan pada lirik dan juga menghapus tanda baca punctuations seperti yang ditunjukkan pada "Gambar 6".

```
# PREPROCESSING DATA
def lowercase(lyric):
    """Mengubah seluruh lirik lagu menjadi huruf kecil"""
    return lyric.lower()

def make_begin(lyric):
    """Memulai dengan __BEGIN__ __BEGIN__ karena
    menggunakan bigram language model"""
    lyrics = lyric.strip().split('\n')
    if len(lyrics) > 0:
        lyrics[0] = '__BEGIN__ __BEGIN__ ' + lyrics[0]
    return '\n'.join(lyrics)

def make_end(lyric):
    """Mengakhiri lirik lagu dengan __END__ sebagai penanda
    bahwa lagu telah selesai"""
    lyrics = lyric.strip().split('\n')
    if len(lyrics) > 0:
        lyrics[-1] = lyrics[-1].replace('\n', '') + ' __END__'
    return '\n'.join(lyrics)

def remove_carriage_feed(lyric):
    """Terkadang data yang ada menggunakan newline jenis '\r\n'
    sehingga kami akan menghilangkannya
    """
    return lyric.replace('\r', '')

def make_newline_token(lyric):
    """Melakukan replace pada newline token dengan __NEWLINE__"""
    return lyric.replace('\n', '__NEWLINE__')
```

Gambar 5. Persiapan Pada Data (a)

```
def make_comma_token(lyric):
    """Melakukan replace pada tanda baca koma dengan __COMMA__"""
    return lyric.replace(',', ' __COMMA__ ')

def make_question_token(lyric):
    """Melakukan replace pada tanda tanya dengan __QUESTION__"""
    return lyric.replace('?', ' __QUESTION__ ')

def remove_parentheses_word(lyric):
    """Di dalam lirik lagu biasanya terdapat kata seperti '(Lirik lagu)'
    Yang biasanya menandakan pengulangan. Kami akan menghilangkan kata
    tersebut
    >>> remove_parentheses_word('Hello World (Hello World)')
    'Hello World '
    """
    return re.sub(r'\([^\)]*\)', '', lyric)

def remove_punctuation(lyric, punctuations='!.,;'):
    """Menghilangkan tanda baca 'punctuations' yang diinginkan
    dari lirik lagu 'lyric'
    """
    for punc in punctuations:
        lyric = lyric.replace(punc, '')
    return lyric
```

Gambar 6. Persiapan Pada Data (b)

4.3. Modelling

Pada tahapan modelling ini, akan di implementasikan algoritma markov chain agar data yang telah melewati tahapan preparation bisa dilakukan training. Disini tim penulis akan melakukan sebuah pemodelan dengan menggunakan markov chain berdasarkan teks lirik yang ada pada data, seperti yang ditunjukkan pada "Gambar 7".

```
# TRAINING / MODEL
class MarkovChain():
    def __init__(self, text):
        self._trained = False
        self.corpus = text
        self.wordfreq = {}

    def train(self):
        """Membangun model menggunakan markov chain berdasarkan teks lirik
        yang telah diberikan
        """
        print('====Currently training====\n')
        for individual_song in self.corpus:
            bigram_tuple_list = self.bigrams(individual_song)
            for i in range(0, len(bigram_tuple_list) - 1):
                current = bigram_tuple_list[i]
                next = bigram_tuple_list[i+1]
                if current not in self.wordfreq:
                    self.wordfreq[current] = {}
                if next not in self.wordfreq[current]:
                    self.wordfreq[current][next] = 0
                self.wordfreq[current][next] += 1
        print('====Training done====\n')
        self._trained = True
```

Gambar 7. Pemodelan Menggunakan Markov Chain

4.4. Evaluation

Pada tahapan ini akan dilakukan evaluasi terhadap lirik lagu yang telah di generate berdasarkan artis. Evaluasinya adalah dengan menanyakan ke 10 orang yang telah dipilih sebelumnya. Adapun yang kami perlihatkan kepada 10 orang tersebut ditunjukkan pada "Gambar 8" dibawah. Dan hasilnya, 9 dari 10 orang menyatakan bahwa itu sangat cocok baik dari segi penyusunan kata dan juga segi kepaduan dari satu lirik tersebut.

Perancangan Sistem Generate Lirik Lagu Berbasis Website Dengan Menggunakan Algoritma Markov Chain

```
===== PAPINKA =====
Currently training...
Training done
cinta ku cinta kepadamu
maafkan aku tak akan pernah mampu
tuk bisa dapat
sembuhkan luka hatiku
kamu pencuri hatiku
dan buat aku terjatuh
gayamu dan lembut suaramu
membuatku semakin suka kepadamu
tergila gila padamu
sungguh terluka hatiku
'ku sudah tak mungkin untuk bersama
kau pergi tinggalkan diriku
tinggalkan aku
```

Gambar 8. Lirik Lagu yang Telah di Generate

4.5. Deployment

Setelah melalui tahapan diatas kemudian model akan disimpan dan kemudian akan di deploy menggunakan HTML dan CSS untuk pengembangan frontend dan framework flask untuk pengembangan backend seperti yang ditunjukkan pada "Gambar 9" dibawah. Tujuan dari deploy ini agar orang lain bisa menggunakannya untuk membuat sebuah lirik lagu baru berdasarkan artis.

```
from flask import Flask, render_template, jsonify
from random import choice
import pickle

app = Flask(__name__)

class DatabaseHandler(DatabaseHandler):
    def find_class(self, module, name):
        if name == "MarkovChain":
            from markov_chain import MarkovChain
            return MarkovChain
        return super().find_class(module, name)

markov_chain_model = DatabaseHandler("markov_chain/markov_chain_artist_model.pkl", "r").load()
artist_list = list(markov_chain_model.keys())

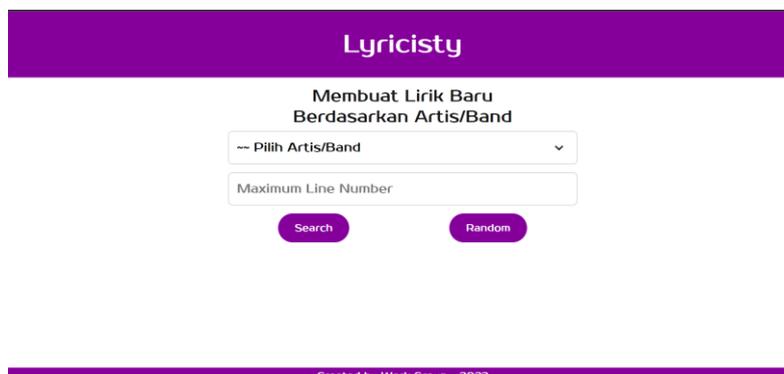
@app.route("/")
@app.route("/index")
def index():
    return render_template("index.html", artists=artist_list)

@app.route("/artist/generate/<string:artist/>(<int:line_number>")
def generate_artist(line_number=None, artist=""):
    if artist in markov_chain_model:
        markov_artist = markov_chain_model[artist]
        if not markov_artist.is_trained():
            markov_artist.train()
        return jsonify({'artist': ' '.join(w.capitalize() for w in artist.split()), 'lyric': markov_artist.generate(line_number)})
    return jsonify({'error': 'Artist not exist'})

@app.route("/artist/generate_random/<int:line_number>")
def generate_random(line_number=None, method="GET"):
    artist = choice(artist_list)
    markov_artist = markov_chain_model[artist]
    if not markov_artist.is_trained():
        markov_artist.train()
    return jsonify({'artist': ' '.join(w.capitalize() for w in artist.split()), 'lyric': markov_artist.generate(line_number)
})
```

Gambar 9. Penggunaan Framework Flask

Pada "Gambar 10" dibawah ini ditunjukkan sebuah UI website dari Lyricisty yang telah didesain dan siap digunakan oleh orang-orang. Dengan memilih salah satu pilihan artis yang tersedia dan kemudian menekan tombol Search maka, akan muncul sebuah lirik yang ditunjukkan pada "Gambar 11". Jika menekan tombol Random maka sistem akan melakukan generate dari sembarangan artis.



Gambar 10. UI Website Lyricisty



Gambar 11. Hasil dari Generate Lagu

5. KESIMPULAN

Penelitian ini memiliki tujuan untuk membuat sebuah lirik lagu yang baru berdasarkan artis-artis dengan menggunakan NLP dan juga algoritma markov chain. Data yang digunakan adalah sebanyak 726 data dengan 3 atribut yaitu nama_artis, judul_lagu dan juga lirik_lagu. Dengan melakukan evaluasi terhadap 10 orang, 9 dari mereka menganggap lirik lagu yang telah di generate itu sangat cocok dan memiliki kesatuan yang padu dari lirik. Hasil dari penelitian ini kemudian di deploy ke sebuah website agar orang-orang bisa menggunakannya.

Kekurangan dari penelitian ini adalah ketidak lengkapnya artis yang ada karena waktu dan juga banyaknya orang di dalam penelitian ini sangat berpengaruh untuk mendapatkan datanya. Untuk penelitian kedepannya diharapkan untuk bisa mendapatkan data yang lebih besar lagi daripada penelitian ini dan juga kelengkapan dari artis-artisnya.

DAFTAR RUJUKAN

- [1] S. Huda and S. Komsiah, "Analisis Semiotik Makna Motivasi Meraih Mimpi Pada Lirik Lagu ' Manusia Kuat ' Karya Tulus," *J. Ilmu Komun. Progress.*, vol. 3, no. 2, pp. 129–142, 2022, doi: <http://dx.doi.org/10.52429/progressio.v3i2.892>.
- [2] I. Syahputra, "Cara Membuat Lirik Lagu Indonesia Sendiri yang Bagus dengan Mudah," *Belajar Musik*, 2021. <https://www.belajarmusiku.com/2015/02/cara-membuat-lirik-lagu-indonesia-sendiri-yang-bagus-dengan-mudah.html> (accessed Nov. 30, 2022).
- [3] N. A. Pramesthi, "Apa Itu AI? Pengertian dan Kegunaannya di Berbagai Industri," *FORTUNE INDONESIA*, 2022. <https://www.fortuneidn.com/tech/nadia/apa-itu-ai-pengertian-dan-kegunaannya-di-berbagai-industri> (accessed Nov. 30, 2022).
- [4] H. E. Rosyadi, F. Amrullah, R. D. Marcus, and R. R. Affandi, "Rancang Bangun Chatbot Informasi Lowongan Pekerjaan Berbasis Whatsapp dengan Metode NLP (Natural Language Processing)," *BRILIANT J. Ris. dan Konseptual*, vol. 5, no. 3, pp. 619–626, 2020, doi: <http://dx.doi.org/10.28926/briliant.v3i4.487>.
- [5] F. S. Riedel, "Investigating the Influence of the Coronavirus on 2020s Music Charts," *Berliner Hochschule für Technik*, 2022.
- [6] Mustakim, I. Kamila, A. Ramadhan, and A. Irwandi, "Implementasi Algoritma Markov Chains untuk Prediksi Kejadian Bencana Alam di Provinsi Riau," in *Seminar Nasional Teknologi Informasi, Komunikasi dan Industri (SNTIKI-10)*, 2018, no. November, pp. 2579–5406.
- [7] N. Hidayati, P. T. Pungkasanti, and N. Wakhidah, "Prediksi Bencana Alam di Kota Semarang Menggunakan Algoritma Markov Chains," *J. Sains dan Inform.*, vol. 7, no. 1,

- pp. 107–116, 2021, doi: 10.34128/jsi.v7i1.283.
- [8] A. S. Oktriwina, "NLP: Kecerdasan Buatan yang Bantu Komputer Pahami Bahasa Manusia," *Glint*, 2021. <https://glints.com/id/lowongan/natural-language-processing-adalah/#.Y4hRk3bP3IU> (accessed Dec. 01, 2022).
- [9] R. Fachrizal, "Apa Itu Natural Language Processing (NLP) dan Apa Saja Contohnya?," *INFO KOMPUTER*, 2021. <https://infokomputer.grid.id/read/122845367/apa-itu-natural-language-processing-nlp-dan-apa-saja-contohnya?page=all> (accessed Dec. 01, 2022).
- [10] Trivusi, "Natural Language Processing (NLP): Pengertian, Metode, dan Manfaatnya," *Trivusi*, 2022. <https://www.trivusi.web.id/2022/08/natural-language-processing.html> (accessed Dec. 01, 2022).
- [11] G. F. Novindri and P. Ocsa Nugraha Saian, "Implementasi Flask Pada Sistem Penentuan Minimal Order Untuk Tiap Item Barang Di Distribution Center Pada PT XYZ Berbasis Website," *J. Mnemon.*, vol. 5, no. 2, pp. 80–85, 2022, doi: 10.36040/mnemonic.v5i2.4670.