

Generador de versos musicales por medio de procesamiento de lenguaje natural y redes neuronales.

Trabajo Terminal No. _____ - _____

*Alumnos: Espinosa de los Monteros Lechuga Jaime Daniel, Nava Romo Edgar Adrián, *Salgado Gómez*

Alfredo Emilio

Directores: Kolesnikova Olga, López Rojas Ariel

e-mail: ttlyrics.escom@gmail.com

Resumen - El presente Trabajo Terminal propone el desarrollo de una aplicación web con interfaz gráfica alojada en la nube que permita la generación de versos musicales con una correcta semántica a través de procesamiento de lenguaje natural y redes neuronales donde a partir de una o más palabras proporcionadas por el usuario, tendencia de algún artista o en su defecto algún género;

Palabras clave - Analizador de Semántica, Aplicación Web, Lector de Semántica, Técnicas de Inteligencia Artificial

1. Introducción

En la actualidad la industria musical obtiene ganancias a través de la creación y divulgación de la música de manera física y digital, dejando que aficionados y emprendedores de la música no tengan oportunidad de avanzar en su carrera. La creación de nuevas letras para sus canciones puede ser un gran obstáculo, por lo que nuestra propuesta implica la utilización de nuevas tecnologías que permitan la generación de letras para sus canciones.

La generación de texto es una de las tareas más populares y desafiantes en el área de procesamiento del lenguaje natural. Recientemente, hay una gran cantidad de trabajos (Generating Text with Recurrent Neural Networks[1], Convolutional Neural Networks for Sentence Classification.[2]) los cuales propusieron generar texto utilizando redes neuronales de atención jerárquica, redes neuronales recurrentes y redes neuronales convolucionales. Sin embargo, la mayoría de los trabajos actuales solo se enfocan en generar una o varias oraciones, ni siquiera un párrafo largo, y mucho menos una canción completa.

Las letras de canciones, como un tipo de texto, tienen algunas características propias. Las cuales suelen tener ritmos, en su mayoría no son muy cortas y tienen algunos patrones de repetición. Estas características hacen que generar letras sea mucho más difícil que generar textos normales.

La mayoría de los trabajos actuales sobre generación de letras vienen con muchas condiciones, como dar una pieza de melodía (Automatic Generation of Melodic Accompaniments for Lyrics[3]), o solo generar un tipo específico de letra (Conditional Rap Lyrics Generation with Denoising Autoencoders[4]). Sin embargo, la generación de letras automáticamente dado un estilo y un tema es un asunto poco trabajado. Por lo tanto, planeamos centrarnos en este nuevo problema. Estamos interesados en ver si nuestro modelo propuesto puede aprender diferentes características para diferentes estilos, distinguir entre varios temas y generar letras con sentido y coherencia.

Actualmente, en el mercado se encuentran tres aplicaciones web que tienen una funcionalidad similar a la propuesta en este Trabajo Terminal:

- These lyrics do not exist
- Bored humans - lyrics_generator
- DeepBeat

En la tabla 1, la cual se muestra a continuación, se muestran las características de las aplicaciones web similares a nuestra propuesta:

SOFTWARE	CARACTERÍSTICAS	PRECIO EN EL MERCADO
These lyrics do not exist	Genera letras completamente originales de varios temas,hace uso de IA para generar coros y versos originales; se puede escoger el tema principal de la letra, género musical e incluso el estado de ánimo al que iría dirigido.	Gratuito (Contiene Anuncios)
Boredhumans - lyrics_generator	La IA fue entrenada con una base de miles de letras para escribir una canción nueva. La letra que crea es única y no una copia de alguna que exista actualmente; sin embargo no permite customizar la letra.	Gratuito
DeepBeat	Es una IA generadora de letras de música enfocada principalmente en el género rap. Si una línea no es del agrado se puede sustituir por alguna de las otras propuestas de las que ofrece.	Gratuito

Tabla 1. Resumen de productos similares.

2. Objetivo

Objetivo General

Desarrollar una aplicación web que utilice un módulo de procesamiento de lenguaje natural y lea la semántica a través de redes neuronales alojadas en la nube, esto es para generar versos completamente nuevos con patrones de repetición y coherencia basados en su género musical, una o más palabras proporcionadas por el usuario, así como alguna tendencia a un artista.

Objetivo Específico

- Crear una interfaz intuitiva para la visualización del verso musical generado.
- Preprocesamiento de información
- Dataset para efecto de entrenamiento de la red semántica
- Módulo de analizador de semántica
- Entrenamiento de la red neuronal

3. Justificación

El crear nuevas composiciones musicales puede llegar a ser muy difícil, estresante e incluso agotador para cualquier aficionado o experto en este medio, esto se debe a la falta de creatividad de quien lo quiera realizar. En ocasiones se pueden contratar servicios particulares para la producción de la letra de una canción, sin embargo, puede ser muy costoso y en ocasiones el resultado final no alcanza a llenar las expectativas de la inversión que se hace; por ende se pretende crear una herramienta para estudiantes, aficionados, expertos en música o cualquier persona interesada en este rubro, a los que se les dificulta componer nuevas letras musicales.

Normalmente las canciones creadas por el humano, tienden a estar compuestas por patrones de acuerdo al género musical, algunos ejemplos de estos patrones pueden ser las rimas, enunciados, frases cortas y una semántica correcta, estos patrones pueden ser encontrados por medio de procesamiento de lenguaje natural y una investigación profunda en la composición de letras de estos géneros.

Nos proponemos orientar la solución en un entorno de nube, donde la información de configuración, servicios y datos necesarios pueden mantenerse de manera independiente a la implementación, facilitando la adaptación y flexibilidad de la plataforma.

Nuestro proyecto ayudará al usuario a través de herramientas como el procesamiento de lenguaje natural, redes neuronales, machine learning y servidores en la nube, generando letras de canciones únicas que podrá guardar en un archivo de texto, permitiéndole obtener inspiración para poder crear o complementar sus propias canciones sin la necesidad de tener que empezar desde cero.

Actualmente existen proyectos similares mostrados anteriormente en la Tabla 1, sin embargo, nuestra propuesta es generar letras musicales con métodos y tecnologías distintas a los que se usaron en los proyectos mencionados anteriormente, esto es, que aunque se utilicen los mismos géneros musicales, se tendrán resultados completamente diferentes con propuestas distintas e innovadoras.

En el desarrollo de este proyecto haremos uso de los conocimientos adquiridos durante el transcurso de la carrera. Se utilizarán técnicas de diseño de proyectos aprendidas en el curso de Ingeniería de Software, aplicaremos los conocimientos de programación adquiridos en unidades de aprendizaje como Inteligencia Artificial, Procesamiento de Lenguaje Natural, Web Application Development, Programación Orientada a Objetos, Análisis de Algoritmos, así como técnicas de construcción de documentos y análisis de semántica vistas en “Análisis y Diseño orientado a Objetos” y “Comunicación Oral y Escrita”.

4. Productos o Resultado esperados

- Interfaz web para el usuario.
- Limpieza de Base de Datos.
- Integración de las redes neuronales con el procesamiento del lenguaje natural.
- Documentación Técnica
- Manual de Usuario



Figura 1. Arquitectura del sistema

5. Metodología

Para el desarrollo de este trabajo terminal se utilizará la metodología ágil Scrum, se decidió usar debido a que este es un proceso de gestión el cual reduce la complejidad en el desarrollo de productos para satisfacer las necesidades de los clientes. Además permite trabajar de manera más eficiente colaborativamente, es decir, en equipo, para obtener el mejor resultado posible.

Ken Schwaber y Jeff Sutherland [5] explican Scrum de una manera clara y simple. Dicen que no es una colección de partes y/o componentes definidos de manera prescriptiva, sino que está basado en un modelo de proceso empírico, basado en la autoorganización de los equipos los cuales logran lidiar con lo imprevisible, resolviendo los problemas complejos inspeccionándolos y adaptándose continuamente.

Scrum contiene los siguientes eventos:

- Planificación del Sprint (Sprint Planning)
- Scrum Diario (Daily Scrum)
- Revisión del Sprint (Sprint Review)
- Retrospectiva del Sprint (Sprint Retrospective)

Estos eventos existen con el fin de establecer una regularidad y minimizar la necesidad de reuniones no definidas en Scrum. Estos eventos son bloques de tiempo (time boxes), de tal forma que todos cuentan con una duración máxima.

También se definen los siguientes artefactos:

- Lista de Producto (Product Backlog)
- Lista de Pendientes del Sprint (Sprint Backlog)
- Incremento (Increment)

Los artefactos en Scrum se definen para así fomentar la transparencia de la información de tal manera que todos los involucrados tengan el mismo entendimiento de que es lo que se está llevando a cabo, además de que nos crean oportunidades para realizar inspecciones y adaptaciones.

Ya que se nos permite crear Sprints, los cuales son ciclos breves de un mes o menos con diferentes fases, en las cuales al final de cada ciclo se define una fecha para la entrega de una versión del producto deseado. Debido a que se trata de una versión, no se indica la finalización del proyecto, sino que habrá un mantenimiento constante para que se obtenga un producto final óptimo.

En virtud de que con el uso de Scrum se logra la integración de todas las partes involucradas en el proyecto, la administración y participación es sencilla y fácil de manejar para todas las etapas. Asimismo se cuenta con un registro de las labores realizadas y se le da un seguimiento. Igualmente nos proporciona una respuesta rápida a los cambios, así como la implementación de pruebas funcionales durante el proceso.

6. Cronograma

Se agrega anexo de cronogramas al final del documento.

7. Referencias

[1] I. Sutskever, J. Martens, G. Hinton, Universidad de Toronto (2011). Generating Text with Recurrent Neural Networks. 10 de Octubre 2020 de https://icml.cc/2011/papers/524_icmlpaper.pdf

[2] Y. Kim, Universidad de Nueva York (Septiembre 2014). Convolutional Neural Networks for Sentence Classification. Recuperado el 10 de Octubre 2020 de <https://arxiv.org/pdf/1408.5882.pdf>

[3] K. MonTeith, T. Martinez D. Ventura (2012). Automatic Generation of Melodic Accompaniments for Lyrics. 10 de Octubre 2020 de <http://computationalcreativity.net/iccc2012/wp-content/uploads/2012/05/087-Monteith.pdf>

[4] Nikola I. Nikolov, E. Malmi, Curtis G. Northcutt, L. Parisi (Abril 2020). Conditional Rap Lyrics Generation with Denoising Autoencoders. 10 de Octubre 2020 de <https://arxiv.org/pdf/2004.03965.pdf>

[5] K. Schwaber, J. Sutherland (2016, julio). La Guía Definitiva de Scrum: Las Reglas del Juego. 10 de Octubre 2020 de <https://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-Spanish.pdf#zoom=100>

8. Alumnos y Directores

Espinosa de los Monteros Lechuga Jaime Daniel. - Alumno de la carrera de Ing. en Sistemas Computacionales en ESCOM, Especialidad Sistemas, Boleta: 2016630100, Tel. 5536517168, email jamesdlechu@gmail.com

Firma: _____

Nava Romo Edgar Adrian. - Alumno de la carrera de Ing. en Sistemas Computacionales en ESCOM, Especialidad Sistemas, Boleta: 2016630492, Tel. 5543925179, email edgar.adrian97@gmail.com

Firma: _____

Salgado Gómez Alfredo Emilio. - Alumno de la carrera de Ing. en Sistemas Computacionales en ESCOM, Especialidad Sistemas, Boleta: 2014131177, Tel. 5538621710, email alfredoe.sgomez@gmail.com

Firma: _____

Kolesnikova Olga. - Doctorado en Ciencias de la Computación. Maestría en Lingüística. Tel. x, email kolesolga@gmail.com

Firma: _____

López Rojas Ariel. - Ingeniero electrónico en comunicaciones. Maestría en Ciencias con especialidad en negocios. Tel. x, email arilopez@ipn.mx

Firma: _____

CARÁCTER: Confidencial FUNDAMENTO
LEGAL: Artículo 11 Fracc. V y Artículos 108, 113
y 117 de la Ley Federal de Transparencia y Acceso
a la Información Pública. PARTES
CONFIDENCIALES: Número de boleta y teléfono.

Nombre del alumno: Espinosa de los Monteros Lechuga Jaime Daniel

Título del Trabajo Terminal: Generador de versos musicales por medio del procesamiento de lenguaje natural y redes neuronales.

[illegible]

Nombre del alumno: Edgar Adrián Nava Romo

Título del Trabajo Terminal: Generador de versos musicales por medio del procesamiento de lenguaje natural y redes neuronales.

[illegible]

Título del Trabajo Terminal: Generador de versos musicales por medio del procesamiento de lenguaje natural y redes neuronales.

[illegible]