

Generador de versos musicales en el idioma inglés por medio de procesamiento de lenguaje natural y redes neuronales

Trabajo Terminal No. 2020 - B002

*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 - Artistas suelen frustrarse durante el proceso creativo para la generación de una letra musical, en consecuencia, pueden llegar a abandonar sus sueños y aspiraciones, el presente Trabajo Terminal propone el desarrollo de una aplicación web intuitiva alojada en la nube que permitirá la generación de versos de un género musical en inglés con semántica correcta a través de procesamiento de lenguaje natural y redes neuronales.

Palabras clave - Analizador de semántica, aplicación web, generador de versos, 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 (Bourreau and Gensollen 2006 [1]), dejando que aficionados y emprendedores de la música no tengan oportunidad de avanzar en su carrera por falta de creatividad, tiempo y/o recursos, haciendo que la creación de nuevas letras para sus canciones sea un gran obstáculo, nuestra propuesta implica la utilización de nuevas tecnologías que permitan la generación de letras para integrar con 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 gran cantidad de trabajos (Generating Text with Recurrent Neural Networks [2], Convolutional Neural Networks for Sentence Classification [3]) los cuales propusieron generar texto utilizando 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, estas se constituyen de rimas, versos, coros y en algunos casos, patrones de repetición. Coro se refiere a la parte de una canción que se repite sin modificaciones dentro de la misma después de un verso. Mientras que en el verso suelen cambiar una o varias líneas que lo componen. Estas características hacen que generar letras musicales sea mucho más difícil que generar textos normales.

La mayoría de las investigaciones actuales sobre generación de letras vienen con muchas condiciones, como dar una pieza de melodía (Automatic Generation of Melodic Accompaniments for Lyrics [4]), o solo generar un tipo específico de letra (Conditional Rap Lyrics Generation with Denoising Autoencoders [5]). Sin embargo, la generación de letras por medio de inteligencia artificial 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 el modelo propuesto puede aprender diferentes características en un género musical y generar letras que sean acorde a este.

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

- These lyrics do not exist.
- Bored humans - lyrics_generator.
- DeepBeat.
- Premium Lyrics.

En la Tabla 1 que se presenta a continuación, se muestran las características de aplicaciones web similares y comparándolas con nuestra propuesta:

SOFTWARE	CARACTERÍSTICAS	PRECIO EN EL MERCADO
These lyrics do not exist	Aplicación web que 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 irá dirigido.	Gratuito (Contiene Anuncios)
Boredhumans - lyrics_generator	Aplicación web en el que la IA fue entrenada con una base de datos con miles de letras para generar letras de canciones totalmente nuevas. La letra que crea es única y no una copia de alguna que exista actualmente, sin embargo, no permite customizar la letra.	Gratuito
DeepBeat	Aplicación web que por medio de IA genera 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
Premium Lyrics	Aplicación web que proporciona versos compuestos en distintos idiomas por artistas independientes que se escogen manualmente de acuerdo a su originalidad y calidad.	3\$ a 75\$ Dólares por letra musical
Nuestra propuesta	Aplicación web que haciendo uso de una IA va a generar letras musicales originales a partir de un género musical en exclusivo, lo que asegurará un resultado final con coros y versos distintos cada vez que se utilice.	Gratuito

Tabla 1. Resumen de productos similares comparados con nuestra propuesta.

2. Objetivos

Objetivo General

Crear una herramienta de apoyo para estudiantes o aficionados interesados en este rubro que se les dificulte componer nuevas letras musicales de un solo género musical debido a la carencia de creatividad, la falta de conocimientos en la estructura del género o que no tengan inspiración suficiente para poder crear nuevas canciones, esto con el fin de impulsar la carrera de futuros artistas en la industria musical que no tengan los suficientes recursos para poder contratar servicios particulares de compositores.

Objetivos Específicos

- Generar un conjunto de datos (dataset) con letras musicales de un género musical para efecto de entrenamiento de la red semántica.
- Hacer uso de alguna herramienta de aprendizaje automático (machine learning) e implementar su uso en la nube para ayudar a procesar las letras musicales de un género en específico.
- Implementar un módulo analizador de semántica para entrenar redes neuronales.
- Desarrollar una interfaz web intuitiva en versión prototipo que utilice una aplicación web alojada en la nube para la visualización del verso musical generado a partir de un género.

3. Justificación

El crear nuevas composiciones musicales puede llegar a ser muy difícil, estresante e incluso agotador para cualquier aficionado o incluso algunos expertos en este medio, esto se debe a la falta de creatividad y/o tiempo de quien lo quiera realizar [6]. 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 o cualquier persona interesada en este rubro que se les dificulte componer nuevas letras musicales.

Normalmente las letras musicales creadas por el humano, tienden a estar compuestas por patrones de acuerdo al género musical [7]. Algunos ejemplos de estos patrones pueden ser las rimas, enunciados, frases cortas y que tengan una semántica correcta, estos pueden ser encontrados por medio de procesamiento de lenguaje natural y una investigación profunda en la composición de letras de estos géneros.

Se eligió el idioma inglés debido a que existe una gran cantidad de bases de datos para procesar, al igual que herramientas y documentación para este idioma.

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

Nuestro proyecto ayudará al usuario utilizando herramientas como el procesamiento de lenguaje natural, redes neuronales, aprendizaje de máquina (machine learning) y servidores en la nube.

Se hará uso de un conjunto de datos (dataset) y herramientas alojadas en la nube (Google Cloud Platform o Amazon Web Services) para procesar estos datos; se pretende utilizar un módulo que encuentre patrones por medio de redes neuronales para analizar la semántica mediante técnicas y herramientas ya existentes de procesamiento de lenguaje natural. Se van a realizar pruebas y experimentos con estas herramientas antes de la implementación (Bert [8], spaCy [9]) para poder generar versos. A su vez se va a desarrollar una interfaz web intuitiva en versión prototipo donde el usuario va a poder utilizar esta herramienta donde se le va a mostrar la letra musical que se va a generar en ese momento.

A diferencia de los proyectos señalados en la Tabla 1 nuestra propuesta es generar letras musicales con métodos y tecnologías distintas a los que se usaron, esto es, aunque se utilicen los mismos géneros musicales, se tendrán resultados completamente diferentes con propuestas distintas.

En el desarrollo de este proyecto haremos uso de los conocimientos adquiridos durante el transcurso de la carrera. Se van a utilizar técnicas de diseño de proyectos aprendidas en el curso de *Ingeniería de Software*, se van a aplicar 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 Resultados esperados

- Una base de datos con letras musicales limpias.
- Integración de las redes neuronales con el procesamiento del lenguaje natural.
- Interfaz web en versión prototipo intuitiva para la visualización del verso musical generado.
- Documentación Técnica.
- Manual de Usuario.

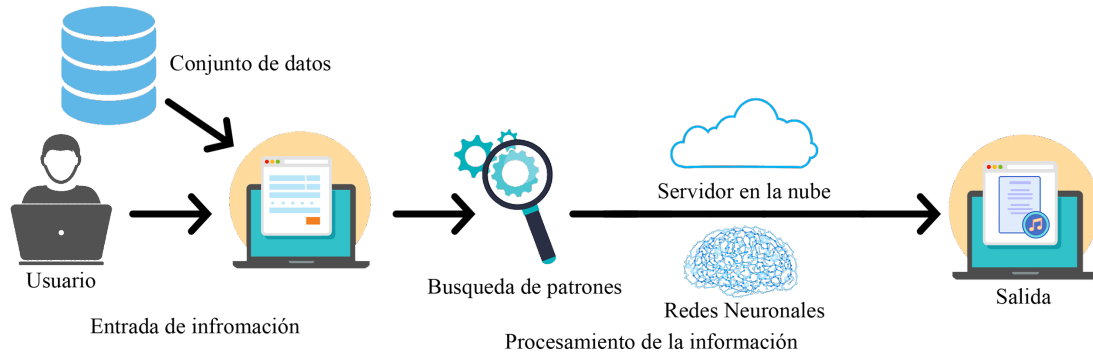


Figura 1. Diagrama del sistema

5. Metodología

Para el desarrollo de este trabajo terminal se utilizará la metodología ágil Scrumban, que combina algunas partes de la metodología Scrum y Kanban, 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 eficiente colaborativamente, es decir, en equipo, para obtener el mejor resultado posible.

Scrumban combina la estructura que utiliza Scrum con los métodos basados en flujo junto con la visualización de Kanban. Es decir, se le permite a los equipos tener la agilidad de Scrum y la simplicidad de Kanban sin tener que utilizar las actualizaciones de roles y es más sencillo de adoptar.

En la siguiente tabla se pueden observar las principales diferencias entre las tres metodologías:

	Scrum	Kanban	Scrumban
Procesos	Iterativo e incremental desarrollando sprints	Continuo	Iterativo e incremental de forma continua desarrollando iteraciones
Personas	Las personas son el centro	Las personas son el pilar	Equipo motivado con personas como pilar y en el centro
Producto	Foco en la efectividad	Foco en la eficiencia	Balance entre efectividad y eficiencia
Organización	Mejora continua del producto	Mejora continua del proceso	Mejora continua del producto y del proceso
Equipo	De 3 a 9 personas	No hay limitaciones	El equipo no requiere de un número específico de integrantes

Roles	Scrum master Product owner Scrum team	Pueden incluir especialistas o integrantes generalizados	No requiere un rol específico
-------	---	--	-------------------------------

Tabla 2. Comparación de metodologías ágiles.

Scrumban utiliza iteraciones y se monitorea con la ayuda de un tablero visual. Las reuniones para planificar se llevan a cabo cuando son necesarias para determinar las tareas a implementar hasta la próxima iteración. Para que estas iteraciones se mantengan cortas, se utiliza un límite de trabajo en progreso (WIP por sus siglas en inglés Work in Progress). Cuando WIP cae por debajo del nivel predeterminado, se establece un activador de planificación bajo demanda para que el equipo sepa cuándo planificar a continuación.

Iteración

En Scrumban, las iteraciones son cortas para garantizar que el equipo pueda adaptarse al entorno cambiante durante el proyecto. La duración de estas iteraciones en este proyecto se medirán como máximo en lapsos de dos semanas.

Priorización

La priorización se da de tal forma que las tareas más importantes se colocan en la parte superior de la tabla de planificación seguidas por las tareas menos importantes.

Antes de llegar al tablero las tareas deben pasar por 3 etapas donde se van depurando las tareas a realizar para largo plazo (1 año), medio plazo (6 meses) y corto plazo (3 meses) siendo de esta última de donde salen las tareas más claras que se pueden completar y que ganan mayor prioridad para la próxima iteración.

Principio de Elección

Cada miembro del equipo elige solo qué tarea de la sección “Tareas Pendientes” va a completar a continuación.

Congelación de Funciones

Se utiliza en Scrumban cuando se acerca la fecha límite del proyecto, significando que sólo pueden trabajar sobre las tareas previamente pensadas sin cabida para implementar nuevas características.

Triaje

Ocurre después de la congelación de funciones, y es el punto donde el gerente del proyecto decide cuáles de las características en desarrollo se completarán y cuáles quedarán sin terminar.

Dado lo anterior, se decidió usar scrumban como metodología y se pretenden desarrollar 4 iteraciones en cada Trabajo Terminal:

Trabajo Terminal 1.

- Se construirá una base de datos acorde al proyecto, la cual contendrá las letras musicales de distintos artistas, así como su género.
- Se realizará una limpieza a los datos extraídos de la base de datos.
- Se seleccionarán e implementarán las herramientas necesarias para trabajar en la construcción de un modelo para la red neuronal.
- Se integrará el modelo generado a la nube para su entrenamiento.

Trabajo Terminal 2.

- Se implementará la interfaz.
- Se integrará el front-end con el back-end.
- Se realizarán las pruebas pertinentes, así como la reingeniería requerida.
- Se desarrollará el manual técnico y el de usuario.

6. Cronograma

Se agrega anexo de cronogramas al final del documento.

7. Referencias

- [1] Chaney, D. (2012). The Music Industry in the Digital Age: Consumer Participation in Value Creation. *International Journal of Arts Management*, (1), pp. 15.
- [2] Sutskever, I., Martens, J., Hinton, G. E. (2011, January). Generating text with recurrent neural networks. In *ICML*.
- [3] Kim, Y. (2014). Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.
- [4] Monteith, K., Martinez, T. R., & Ventura, D. (2012, May). Automatic Generation of Melodic Accompaniments for Lyrics. In *ICCC*, pp. 87-94.
- [5] Nikolov, N. I., Malmi, E., Northcutt, C. G., Parisi, L. (2020). Conditional Rap Lyrics Generation with Denoising Autoencoders. *arXiv preprint arXiv:2004.03965*.
- [6] Baker, F. A. (2015). What about the music? Music therapists' perspectives on the role of music in the therapeutic songwriting process. *Psychology of Music*, 43(1), pp. 122-139.
- [7] Guerrero, J. (2012). El género musical en la música popular: algunos problemas para su caracterización. *Trans. Revista transcultural de música*, (16), pp. 1-22.
- [8] Wang, A., & Cho, K. (2019). Bert has a mouth, and it must speak: Bert as a markov random field language model. *arXiv preprint arXiv:1902.04094*.
- [9] Honnibal, M., & Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing.

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 Adrián. - 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. 5521774791, email kolesolga@gmail.com

Firma: _____

López Rojas Ariel. - Docente ESCOM. Tel. 57-29-6000 Ext. 52032, 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.

Anexo 1

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]

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]

Acuse de recibo por parte de los directores e integrantes del protocolo

“Generador de versos musicales en el idioma inglés por medio de procesamiento de lenguaje natural y redes neuronales”

*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*

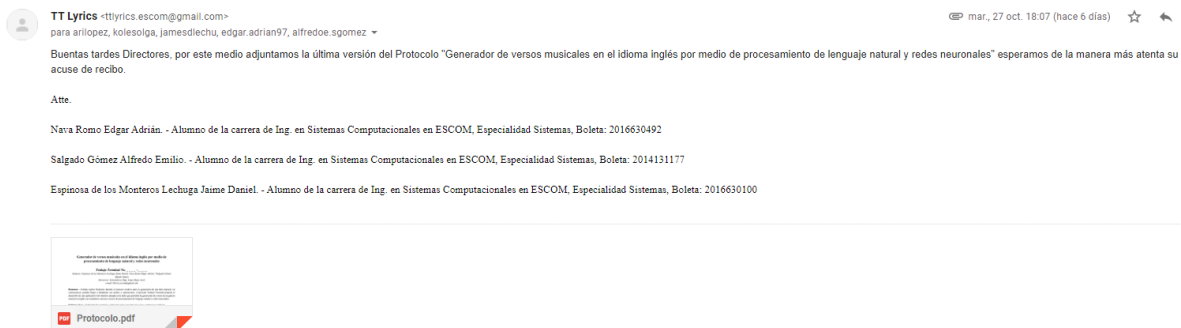


Imagen 1. Correo enviado a los directores donde se presenta la versión final del protocolo entregado.

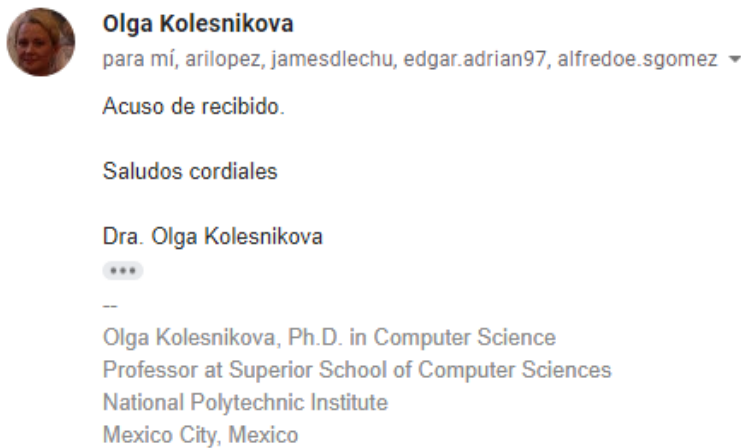


Imagen 2. Acuse de recibo por parte de nuestra directora Kolesnikova Olga.

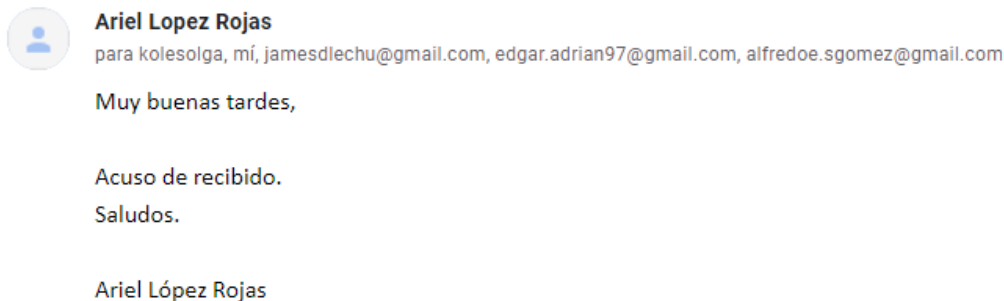


Imagen 3. Acuse de recibo por parte de nuestro director López Rojas Ariel.



Jaime Daniel Espinosa de los Monteros Lechuga

para mí ▼

Acuso de recibido

Atte. Espinosa de los Monteros Lechuga Jaime Daniel

Boleta: 2016630100



Imagen 5. Acuse de recibo por parte del integrante del protocolo Espinosa de los Monteros Lechuga Jaime Daniel.



Adrian Nava <edgaradrian97@gmail.com>

para mí ▼

Acuse de Recibido.

Atte.

Edgar Adrián Nava Romo

Boleta 2016630492

Imagen 5. Acuse de recibo por parte del integrante del protocolo Nava Romo Edgar Adrián.



Alfredo S. Gomez

para mí ▼

Acuse de Recibido.

Atte.

Salgado Gómez Alfredo Emilio

Boleta 2014131177

Imagen 6. Acuse de recibo por parte del integrante del protocolo Salgado Gomez Alfredo Emilio.