

Modelo de literariedad usando redes semánticas y n-gramas

Jonatan Ahumada Fernández

Tesis para el título de Ingeniería de Sistemas

Facultad de Matemáticas E Ingeniería
Fundación Universitaria Konrad Lorenz
Bogotá, Colombia
2022

Contents

1	FORMULACIÓN DEL PROBLEMA	2
1.1	Introducción	2
1.2	Planteamiento del problema	3
1.3	Justificación	4
1.3.1	Palabras clave:	5
1.4	Alcances y delimitaciones:	5
2	OBJETIVO GENERAL	5
3	OBJETIVOS ESPECÍFICOS	5
4	MARCO TEÓRICO	6
4.1	Literariedad	6
4.2	Poética	6
4.3	Linguística	7
4.4	Lingüística Computacional	7
5	MARCO REFERENCIAL	8
6	DISEÑO METODOLÓGICO	9
6.1	Entendimiento del negocio	9
6.2	Entendimiento de los datos	9
6.2.1	Los recursos lexicos	9
6.2.2	La red semántica y similaridad con Saussure	10
6.2.3	Por qué utilizo el Brown Corpus	10
6.3	Preparación de los datos	10
6.4	Modelamiento	10
6.4.1	Selecting a modeling technique (no tengo, estoy traduciendo un modelo cualitativo –investigacion mixta–)	10
6.4.2	Generating a test desing	10
6.4.3	Building the models	11
6.5	Despliegue (los notebooks?? creo que no hay despliegue)	14
7	CONCLUSIONES (Creo que esto se solapa con lo que crisp-dm llama despliegue)	14
8	BIBLIOGRAFÍA	14

1 FORMULACIÓN DEL PROBLEMA

1.1 Introducción

¿Qué constituye la esencia de un texto? ¿Qué diferencia un texto considerado 'literario' de aquél que no lo es? Esta pregunta se ha planteado en áreas como los estudios literarios y la lingüística [3]. Particularmente, la escuela denominada

'formalismo ruso' planteó que el objeto de estudio de la literatura, no *podría* ser la belleza, la relevancia histórica o el valor pragmático de un texto. Más bien, su objeto de estudio *debe* recaer en un aspecto más 'objetivo': su *literariedad*. Como su nombre sugiere, los formalistas se abocaron a formular una definición 'objetiva' y 'concreta' del fenómeno literario y adoptaron los –en ese entonces– modernos métodos de la buyente disciplina de la lingüística.

Siendo este el caso, ¿no es, por consiguiente, factible que un autómatas pueda medir y presentar tales características presuntamente formales con las actuales herramientas informáticas? ¿Cómo se podría traducir la noción de *literariedad* a un algoritmo que pueda ejecutar una máquina?

1.2 Planteamiento del problema

Roman Jakobson propone que la *literariedad* de un texto está dada por dos componentes de lenguaje: la diacronía y la sincronía. Estos elementos fueron expandidos de la teoría lingüística de Saussure. Más tarde, puestos en el contexto del análisis de la poesía, Jakobson renombró esos dos ejes como *metáfora* y *metonímia*, en su texto "Lingüística y poética".

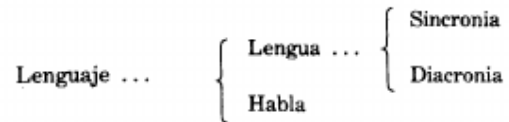


Figure 1: Distinción entre sincronía y diacronía

¿Es posible modelar algorítmicamente tales conceptos? Según Jakobson, en el estudio de la *literariedad* se omite el factor emisor y factor receptor. Tan solo se centra en el mensaje. Representado únicamente a través de un *medio* particular: en este caso, la palabra escrita. Es, por lo tanto, *factible* que un autómatas pueda medir y presentar tales características.

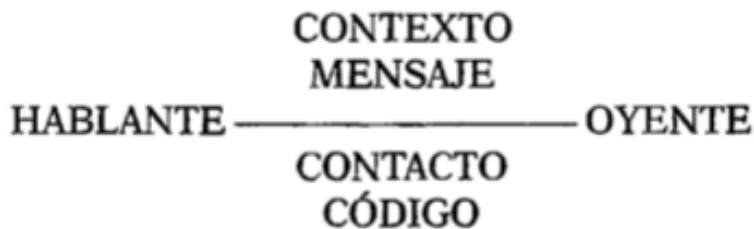


Figure 2: Factores de comunicación de Roman Jakobson

Saussure ofrece ya un modelo cualitativo muy bien esbozado en teoría, que es el que luego Jakobson utilizará para definir la literariedad. Sin embargo, aunque existe un planteamiento cualitativo del problema, no se halló en la bibliografía consultada un modelo computacional que modelara el concepto y lo implementara.

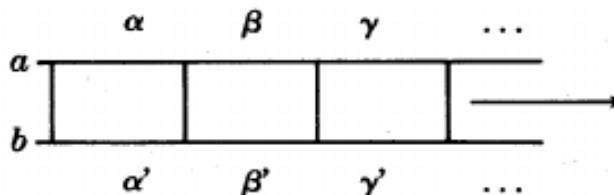


Figure 3: Modelo cualitativo inicial expuesto por Ferdinand De Saussure

Preliminarmente, se puede observar que el modelo de Saussure se fundamenta en una estructura bastante familiar en la computación: la secuencia. Así, el objetivo de este trabajo es modelar e implementar el modelo de *literariedad* de Roman Jakobson utilizando redes semánticas y n-gramas.

1.3 Justificación

Si bien existen infinitas operaciones realizables sobre un texto computarizado, hay pocas que tengan un enfoque humanístico. En este enfoque se busca someter a escrutinio los conceptos del marco teórico, en contraposición a los enfoques 'típicos' –y hoy en día indispensables– de procesamiento de lenguaje: extracción de información, clasificación con base a un modelo predictivo, entre muchos otros [1].

Más aún, dentro de este subconjunto reducido, pocos están guiados por aquello que Gelbukh llama 'la ciencia fundamental'. A saber, la lingüística. En otras palabras, hay un vacío en el campo de la lingüística computacional en lo que se refiere a modelos que procuran cuantificar esta perspectiva.

Tal vacío genera que el estudio académico de la literatura no pueda sustentarse en datos 'duros' o, por lo menos, cuantitativos propias del método científico. Por otro lado, los diversas y posibles formas de calcular la 'creatividad', la 'rima' o la 'belleza' de un texto, propuesto por otros investigadores, pueden considerarse casuísticos, acoplados a las objetivos y circunstancias de cada investigación en particular, desde la perspectiva de la lingüística general.

Así, se necesita un modelo de la *literariedad* que exprese concretamente la metáfora y la metonimia. Bien sea para ampliar las aplicaciones de la lingüística computacional o para someter a escrutinio los planteamientos de la teoría.

En esta investigación se formulará y evaluará un modelo para obtener una medida cuantitativa para el concepto de *literariedad* de Roman Jakobson uti-

lizando redes semánticas y n-gramas. De este modo, la presente investigación respondería a la pregunta ¿Cómo medir computarizadamente la *literariedad* de un texto según el marco de la lingüística de Jakobson?

1.3.1 Palabras clave:

NLP, computational linguistics, literariness, literary theory, poetics, theory of formal method

Área de conocimiento:

Lingüística computacional

1.4 Alcances y delimitaciones:

Para computar una métrica de *literariedad* será necesario comparar un *corpus objetivo* con respecto a un *corpus de referencia*, este último representará el ‘uso corriente de la lengua’. La primera limitación de este trabajo es que no se compilará un corpus propio, sino que partirá de los de acceso libre. La mayoría de estos se encuentran en inglés. Por este motivo, el corpus de referencia más a la mano es WordNet, que al ser una ontología ya contiene las anotaciones necesarias para mi objetivo. A saber, una lista de sinónimos por palabras. Por otro lado, el corpus objetivo no tiene que estar anotado (utilizaré un PlainTextCorpus), pero de algún modo tiene que ser razonable su comparación con el corpus objetivo. Por ejemplo, los resultados del modelo serían muy difíciles de evaluar si la relación entre corpus objetivo y de referencia sobrepasa los 2 siglos, dada la naturaleza fluida de la lengua.

La segunda limitación concierne a la formulación de los algoritmos en sí mismos. Me limitaré a formular los modelos más naive posibles. Por ejemplo, (retomando el ejemplo previo) dada una palabra se considerará un sinónimo todas las palabras listadas como tal en el corpus de referencia, sin considerar los sub-problemas que esto podría conllevar.

En general, el alcance de este proyecto es formular e implementar un modelo general que muestre cómo sería viable implementar el concepto de *literariedad*, sin ahondar en los detalles que se desprenden de cada fase del flujo de NLP (por ejemplo, ¿cómo tokenizar?, ¿Qué peso tendrían las diferentes partes de una oración en el computo final, etc).

2 OBJETIVO GENERAL

Diseñar e implementar un modelo que, dado un corpus de texto, produzca indicadores para el concepto de *literariedad* que plantea Roman Jakobson.

3 OBJETIVOS ESPECÍFICOS

1. Construir el corpus necesario para representar el *eje sincrónico*

2. Diseñar e implementar el algoritmo para calcular la *metáfora* sobre un corpus
3. Diseñar e implementar algoritmo para calcular la *metonimia* sobre un corpus
4. Seleccionar y unir los textos que serán procesados (corpus objetivo) por el algoritmo
5. Correr el algoritmo sobre los corpus objetivo
6. Evaluar el algoritmo de manera cuantitativa y cualitativa

4 MARCO TEÓRICO

4.1 Literariedad

La *literariedad* es, según Jakobson, la cualidad de un objeto literario en cuanto tal. Por lo tanto, la *literariedad* no depende de ningún factor extrínseco, como su emisor, su valor histórico, las ventas de tal o cual libro, las citaciones, etc. La *literariedad* se da exclusivamente por atributos propios del fenómeno del lenguaje.

Para analizar la *literariedad*, se deben analizar las dos operaciones más básicas de la conducta verbal: *la selección* y *la combinación*.

1. Selección (ver lingüística sincrónica):

La selección estudia qué palabra selecciona un hablante entre las palabras existentes de la lengua, más o menos similares y hasta cierto punto equivalentes. La selección se basa en la sinonimia o antonimia de una palabra. En otros términos, en su semántica.

1. Combinación (ver lingüística diacrónica):

La combinación estudia el "entramado de la secuencia" de un mensaje. Es decir, el mensaje considerado como una secuencia temporal y/o ordenada de palabras. La combinación se basa en la proximidad o, en otras palabras, en la relación de una palabra con la que la sucede o antecede en un mensaje.

4.2 Poética

La poética procura responder a la pregunta de ¿qué hace que un mensaje (verbal o de otra naturaleza) sea una obra de arte? Lidia principalmente con cuestiones estéticas del lenguaje. Sin embargo, para hacer un análisis exhaustivo, la poética debe hacer uso de la lingüística, puesto que esta última estudia el lenguaje en todo su conjunto. La *literariedad* podría, entonces, considerarse un concepto enmarcado en la poética, porque se preguntará qué hace que un texto sea literario y por qué es distinto de otro que no lo es.

4.3 Lingüística

La lingüística es la ciencia que estudia el lenguaje. Tradicionalmente, esta ciencia se subdivide en las ramas de fonética, fonología, morfología, sintaxis, semántica y pragmática.

La lingüística es un campo de estudio interdisciplinar e involucra disciplinas heterogéneas como la lógica y la neurolingüística. Sin embargo, se considera que hay un núcleo común llamado *lingüística general*.

1. Lingüística General:

Se conoce como lingüística general al paradigma lingüístico establecido por Ferdinand De Saussure, también llamado *modelo diferencial del lenguaje*.

El modelo diferencial se caracteriza porque propone dos ejes principales existentes en todo fenómeno lingüístico: el *eje de sincronía* y el *eje de diacronía*.

Estos dos ejes son la base de lo que Jakobson considera *selección* y *combinación*.

1. Lingüística sincrónica

La lingüística sincrónica se ocupa de las operaciones que realiza un hablante, sean lógicas o psicológicas, para formar un sistema lingüístico. En el marco de esta investigación el *eje sincrónico* se referirá a las posibles palabras que un hablante pudo haber seleccionado para expresar una misma idea.

1. Lingüística diacrónica

La lingüística diacrónica estudia los campos sucesivos en el lenguaje, producidos por la actividad constante del *eje sincrónico*. En la perspectiva de Jakobson, un *mensaje* tiene en sí mismo un eje diacrónico. Tal eje mide la similaridad entre cada término del mensaje entendido como secuencia: "... para decirlo de un modo más técnico: todo secuencia es un símil."

4.4 Lingüística Computacional

Es la intersección entre la computación y la lingüística. Por lo general, se preocupa acerca de cómo procesar automáticamente el lenguaje material, para lo cual genera modelos lingüísticos sobre los que luego se pueden definir operaciones comunes [6].

La lingüística computacional es en sí misma un campo amplio y heterogéneo, pero en términos de este trabajo, me limitaré a señalar una herramienta:

1. NLTK

El Natural Language Toolkit (NLTK) es un módulo de Python que ofrece una interfaz para tareas comunes en la lingüística computacional. La ventaja principal de NLTK es que se considera a sí mismo un *toolkit*. Esto significa que no impone una estructura de procesamiento definida a la vez que ofrece un

extenso abanico de herramientas, tales como: tokenización, filtros, generación de n-gramas, análisis sintáctico de oraciones, entre otras.

1. Corpus

Un corpus es una colección de textos auténticos que pueden ser leídos por una máquina. Estos pueden estructurarse de muchas formas, dependiendo de los objetivos de la investigación [9]. Por ejemplo, pueden ser aislados (una colección arbitraria), categorizados (una colección escogida según algún criterio), temporales (una colección organizada cronológicamente) o solapados (un documento puede pertenecer a varias colecciones) [10]. Además, el formato del corpus varía significativamente de acuerdo al objeto de la investigación. Por ejemplo, si se desea hacer un análisis sintáctico (de la estructura de una oración), se debe hacer un corpus anotado con POS (Part Of Speech tag); para hacer un análisis pragmático se utiliza una anotación pragmática, etc.

5 MARCO REFERENCIAL

El trabajo de Delmonte [2] presenta a SPARSAR, un sistema para calcular automáticamente el estilo de la poesía. SPARSAR funciona sobre sistemas previos del mismo autor, como, por ejemplo, un analizador semántico [7]. Delmonte tiene una larga trayectoria en el modelamiento de conceptos lingüísticos "difíciles", como la prosodia y la rima en términos cuantitativos.

El aporte principal de Delmonte fue su innovación al momento de aplicar herramientas comunes de NLP (tokenizadores, splitters y NER) con el fin de analizar aspectos estilísticos de un texto. Los modelos de Delmonte son muy cercanos a la teoría lingüística y propone soluciones a aspectos complejos del análisis lingüístico. Esta proximidad me llevo a plantearme la pregunta ¿qué otros aspectos del lenguaje valdría la pena modelar que aún no hayan sido abordados desde una perspectiva computacional? Así mismo, Delmonte reporta que hay pocos trabajos en el área con este mismo enfoque. Esta fue una inspiración para explorar más en el tema y ofrecer un enfoque distinto, tal como él lo hizo.

Sin embargo, Delmonte no revela detalles de implementación de sus sistemas en los artículos revisados. Además, sus sistemas tienen un alcance mucho mayor que el dispuesto para este trabajo, por lo que para mayores detalles tuve que referirme a otros trabajos.

El trabajo de [5] establece una métrica para medir el grado de creatividad en la poesía, basándose en qué tanto de la rima se conserva en la traducción de un poema con respecto al original. Tomé de Zuñiga la idea de establecer una métrica para un aspecto tradicionalmente cualitativo (la creatividad). Lo que diferencia este trabajo del de Delmonte, es su aproximación matemática. Particularmente, Zuñiga ofreció una forma naive de calcular similitud en rima, sin necesidad de recurrir a construcciones que requieren de recursos léxicos complejos como una ontología para fonemas, etc.

Por último, el trabajo de [8] es una tesis de pregrado sobre el cálculo del estilo de la poesía desde una perspectiva estadística. Kaplan fue una inspiración para

Delmonte, por lo tanto debía formar parte de mi revisión bibliográfica. El aporte principal a mi trabajo ha sido tener una serie de operaciones como referencia. Por ejemplo, para modelar aspectos ortográficos, sintácticos y fonémicos, entre otros.

6 DISEÑO METODOLÓGICO

6.1 Entendimiento del negocio

(Me pareció que esto no aplica. Discutir con profesor. Propongo esto otro)

Lo otro sería explicar los posibles usos (posibles clientes) de mi modelo

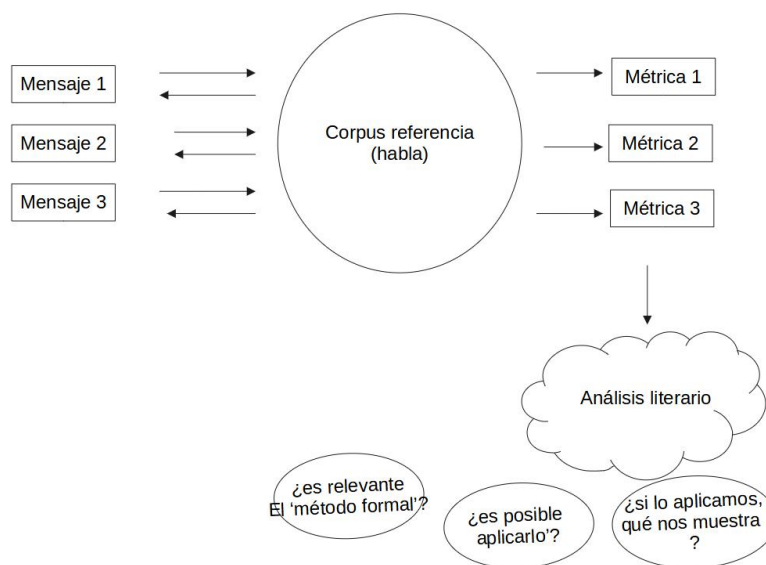


Figure 4: Entradas y salidas del algoritmo

6.2 Entendimiento de los datos

The data understanding phase of CRISP-DM involves taking a closer look at the data available for mining. This step is critical in avoiding unexpected problems during the next phase—data preparation—which is typically the longest part of a project.

6.2.1 Los recursos lexicos

1. Corpus de referencia El corpus de referencia es Wordnet.

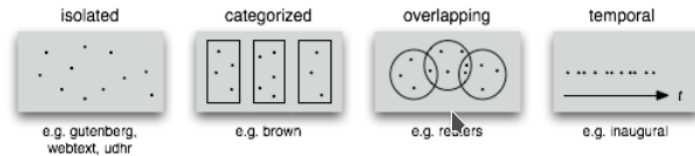


Figure 5: Diferentes estructuras de corpus

2. Corpus objetivo El corpus escogido fue el corpus de Brown porque cumplía con las siguientes criterios:

- (a) La lengua inglesa tiene una correspondiente red semántica
- (b) Esta categorizado, por lo que se espera observar diferencias significativas en el resultado de su procesamiento
- (c) Es fácilmente accesible a través de Python

6.2.2 La red semántica y similaridad con Saussure

6.2.3 Por qué utilizo el Brown Corpus

6.3 Preparación de los datos

Depending on your organization and its goals, data preparation typically involves the f

Merging data sets and/or records

Selecting a sample subset of data

Aggregating records

Deriving new attributes

Sorting the data for modeling

Removing or replacing blank or missing values

Splitting into training and test data sets

6.4 Modelamiento

6.4.1 Selecting a modeling technique (no tengo, estoy traduciendo un modelo cualitativo –investigacion mixta–)

6.4.2 Generating a test desing

- Describing the criteria for "goodness" of a model
- Defining the data on which these criteria will be tested

1. Sampleo de la muestra

- qué textos voy a someter a procesamiento
- por qué escogí estos textos en particular

6.4.3 Building the models

1. Presentacion de las ecuaciones

$$mensaje = \{w_1, w_2, w_3, \dots, w_j\} \quad (1)$$

$$vector\ semantico(w) = \{s_1, s_2, s_3, \dots, s_j\} \quad (2)$$

$$uso(w) = \frac{freq(w)}{freqMedia} \quad (3)$$

$$freqMedia = \mu(freq(corpora\ ref)) \quad (4)$$

$$indice\ metaforico(mensaje) = \sum_i^j \frac{uso(w_i)}{\mu(vector\ semantico(w_i))} \quad (5)$$

$$N = \{n_1, n_2, n_3, \dots, n_j\} \quad (6)$$

$$met(n) = \frac{letras\ iguales}{set(letras(n_i1) + letras(n_i2))} \quad (7)$$

$$indice\ metonimia = \sum_i^j met(n_i) \quad (8)$$

2. Procedimientos para indicadores

3. Índice Metafórico

4. Matriz semántica

5. Matriz de uso

6. Índice Metonímico

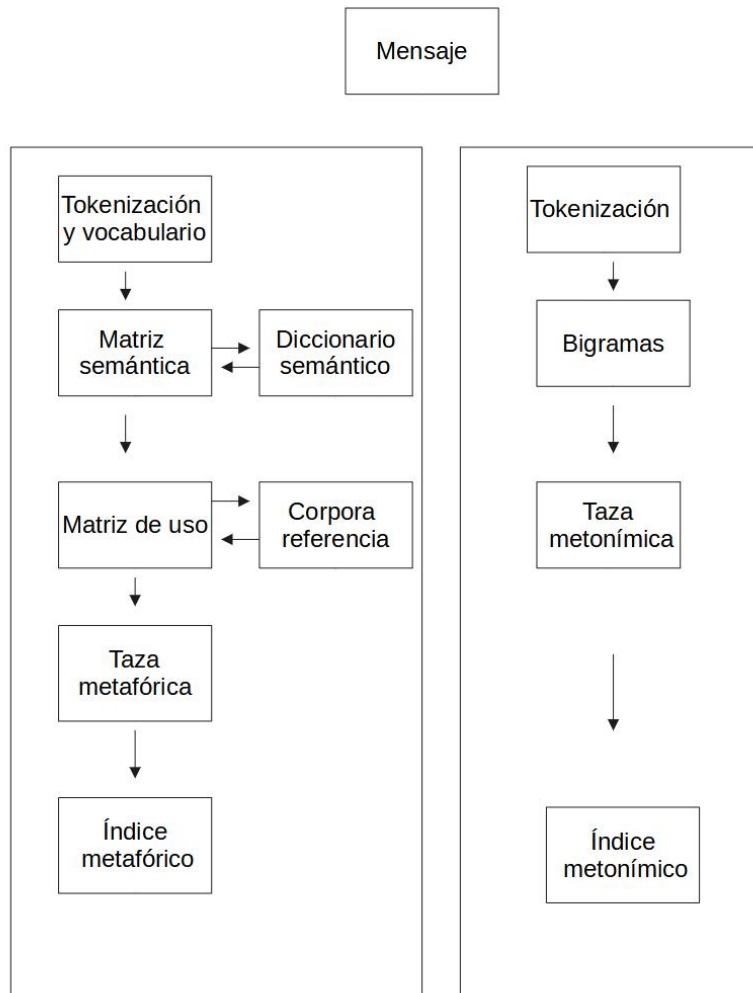


Figure 6: Procesamiento de corpus objetivo

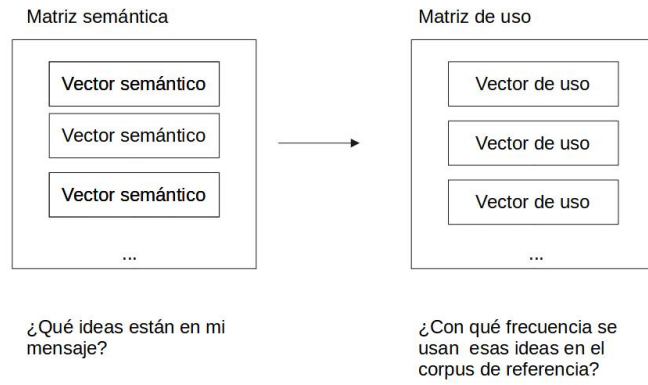


Figure 7: Abstracciones necesarias para el índice metafórico

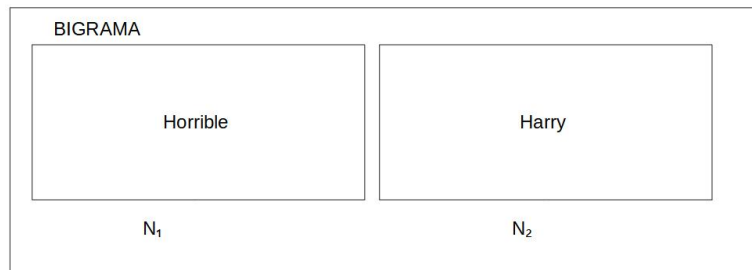


Figure 8: Abstracciones necesarias para el índice metonímico

6.5 Despliegue (los notebooks?? creo que no hay despliegue)

Deployment is the process of using your new insights to make improvements within your organization. This can mean a formal integration such as the implementation of a IBM® SPSS® Modeler model producing churn scores that are then read into a data warehouse. Alternatively, deployment can mean that you use the insights gained from data mining to elicit change in your organization. For example, perhaps you discovered alarming patterns in your data indicating a shift in behavior for customers over the age of 30. These results may not be formally integrated into your information systems, but they will undoubtedly be useful for planning and making marketing decisions.

7 CONCLUSIONES (Creo que esto se solapa con lo que crisp-dm llama despliegue)

8 BIBLIOGRAFÍA

- [1] L. Danlos, “The linguistic basis of text generation,” in Proceedings of the third conference on European chapter of the Association for Computational Linguistics, pp. 1–1, 1987.
- [2] R. Delmonte, “Computing poetry style,” in ESSEM@ AI* IA, pp. 148–155, 2013.
- [3] B. Eijembaum, “La teoría del "método formal",” in Textos de teorías y crítica literarias:(del formalismo a los estudios postcoloniales), pp. 33– 62, Anthropos, 2010.
- [4] R. Jakobson and A. M. G. Cabello, Lingüística y poética. Cátedra España, 1981.
- [5] D. F. Zuñiga, T. Amido, and J. E. Camargo, “Automatic computation of poetic creativity in parallel corpora,” in Colombian Conference on Computing, pp. 710–720, Springer, 2017.
- [6] I. A. Bolshakov and A. Gelbukh, Computational Linguistics: Models, Resources, Applications. Mexico City: Centro de Investigación en Computación, Instituto Politécnico Nacional, 1981.
- [7] R. Delmonte, S. Tonelli, M. A. P. Boniforti, and A. Bristot, “Venses—a linguistically-based system for semantic evaluation,” in Machine Learning Challenges Workshop, pp. 344–371, Springer, 2005.
- [8] D. Kaplan, “Computational analysis and visualized comparison of style in american poetry,” Unpublished undergraduate thesis, 2006.
- [9] Indurkha, N., & Damerau, F. J. “/Handbook of natural language processing/ (Vol. 2)”. CRC Press, 2010
- [10] Bird, Steven, et al. Natural Language Processing with Python. O’Reilly Media, 2009.

References

- [1] Igor A. Bolshakov and Alexander Gelbukh. *Computational Linguistics: Models, Resources, Applications*. Mexico City: Centro de Investigación en Computación, Instituto Politécnico Nacional, 1981.