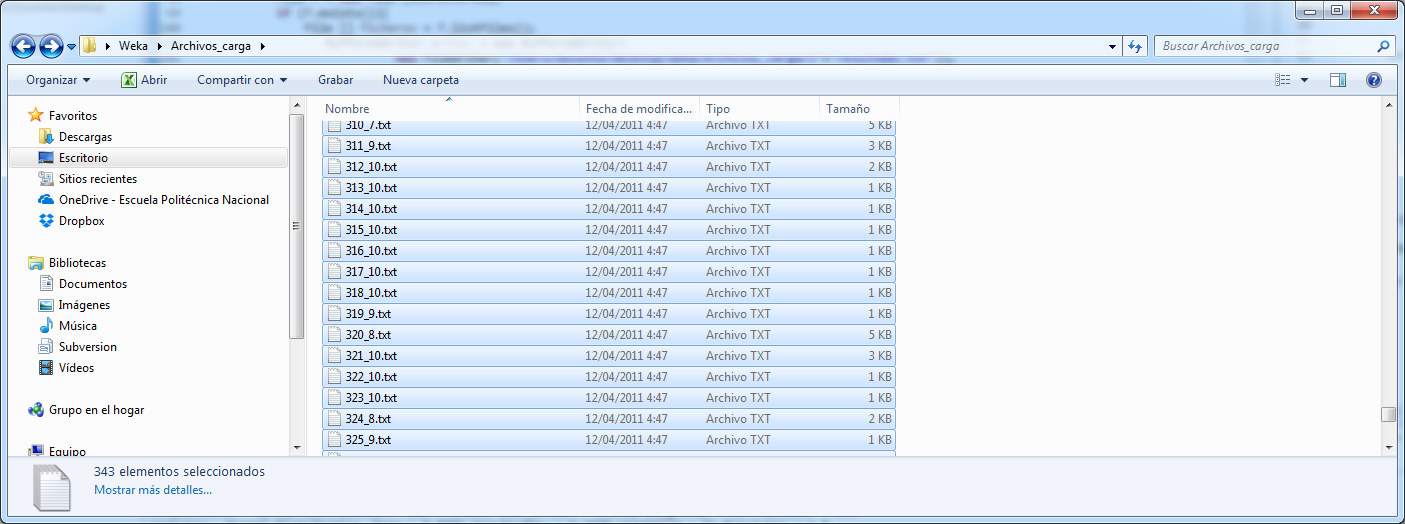
|  |  |
| --- | --- |
| **TEMA** | **Diseño e Implementación de un sistema clasificador de sentimientos** |
| **Estudiante(s)** | **1.Graciela González**  **2.Doris Tutillo** |
| **Fecha** | **20 de enero de 2017** |
| **Profesor** | **MSc. Elisa Mena** |

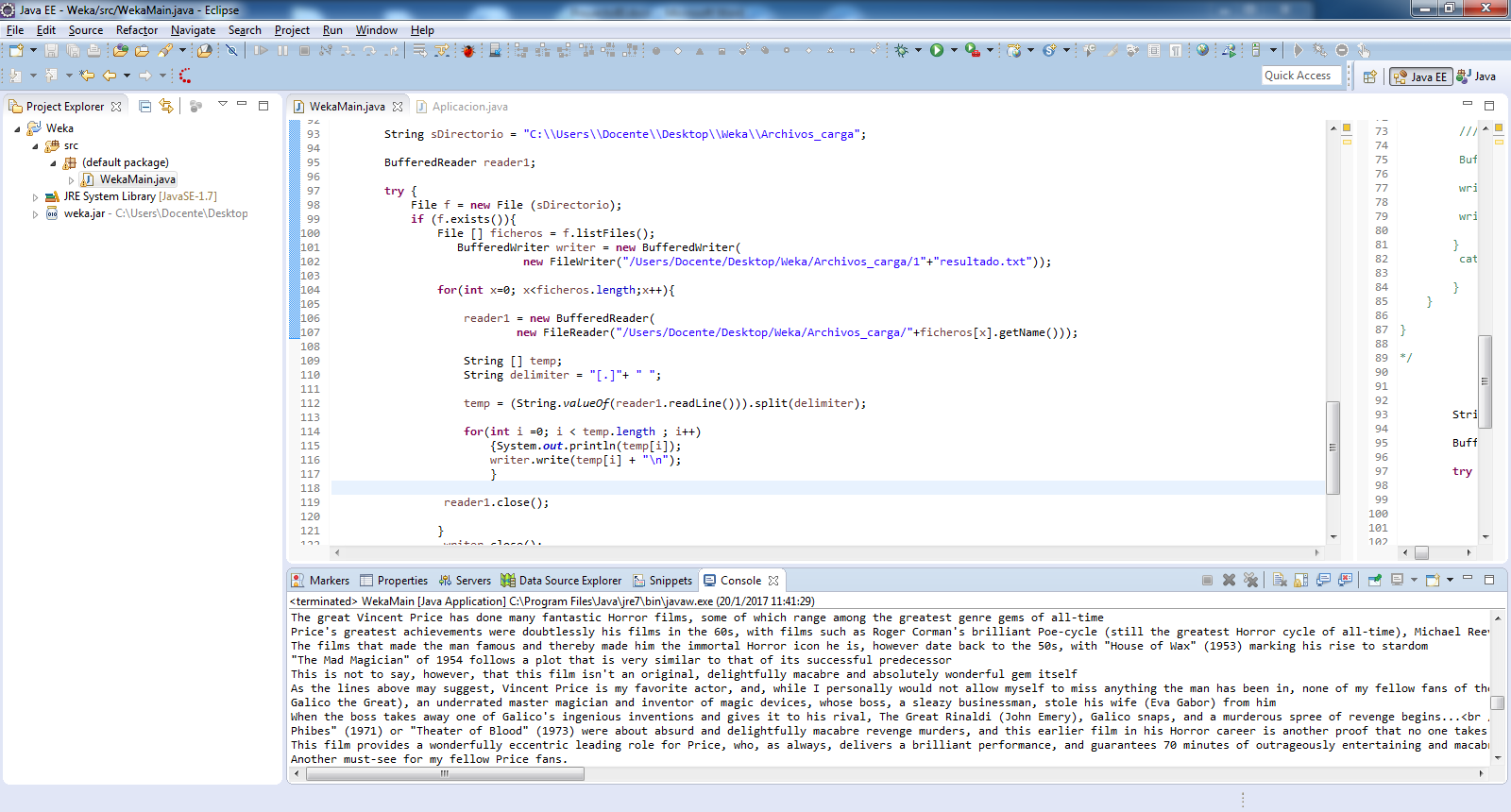
1. **OBJETIVO**
   1. Implementar e investigar el funcionamiento de un clasificador de sentimientos utilizando como base el conjunto de datos de entrenamiento desarrollado.
2. **FASES DEL PROYECTO**
   1. **Determinación de Sentimientos**

La necesidad de evaluar los sentimientos que se expresan a nivel de redes sociales en los últimos años ha crecido, y más aún cuando las empresas que prestan bienes o servicios desean conocer el grado de aceptación por parte de sus clientes, puesto que a través de ellos pueden definir estrategias que permitan mantener o mejorar sus ventas y su posición en el mercado.

Para el presente proyecto, se analizaron las frases o comentarios de un gran conjunto de datos de críticas de películas de la base de datos de películas en Internet (IMDb). [1]

Este conjunto de Datos tiene alrededor de 50.000 registros calificados como positivos y negativos, para lo cual se procedió a recolectar dicha información y subirla a un solo archivo llamado “1resultado”. Siendo el punto “.” el delimitador seleccionado para separar las frases u oraciones.





Dentro del alcance de nuestro proyecto se tomaron en cuenta alrededor de 2114 instancias, dentro de las cuales se incluyeron frases o comentarios que se les etiquetó como neutrales al no tener ningún adjetivo positivo o negativo.

* 1. **Pre procesamiento**

Luego de disponer del conjunto de datos de entrada se los clasificó dependiendo de los siguientes factores:

* **Adjetivo positivo**: Son características positivas que califican al sujeto, se tomaron en cuenta por ejemplo: good, excelent, perfect, beautiful, great.
* **Adjetivo negativo**: Son características negativas que califican al sujeto, se tomaron en cuenta por ejemplo: bad, horrible, terrible.
* Conjunciones: son empleadas para enlazar entre sí las palabras y/o oraciones. Hay dos tipos de conjunciones y la posición que tiene dentro de una oración depende del tipo. Además, hay tres formas de conjunciones. Las conjunciones más comunes son “and,” “but” y “or”.

Conjunciones coordinantes: Este tipo de conjunción se utiliza cuando queremos enlazar dos frases que tienen el mismo valor, tales como: and, but, however, or, so, then, therefore, yet.

Conjunciones subordinantes: Se utiliza este tipo de conjunción cuando una de las frases depende de la otra (frase subordinada). La frase subordinada no tiene sentido sin la otra. La mayoría de conjunciones son subordinantes. Tales como: although, as, after before, if, since, so that, until, when.

Para definir la clase se tomaron en cuenta 3 clasificaciones

* + Positivo
  + Negativo
  + Neutro
* Negaciones: Se realizó la validación de los términos not, dont, didnt, los mismos que denotan negación
* Verbo Positivo: se clasificó a los verbos que denotan una acción positiva como: love, enjoy.
* Verbo Negativo: se clasificó a los verbos que denotan una acción negativa como: hate, die.

Tomando en cuenta que algunas columnas pueden contener tal cantidad de valores que el algoritmo no puede identificar con facilidad patrones de interés en los datos para crear un modelo a partir de los mismos, se decide utilizar valores numericos que expresen cuantas veces están presentes en la frase o comentario. [2]

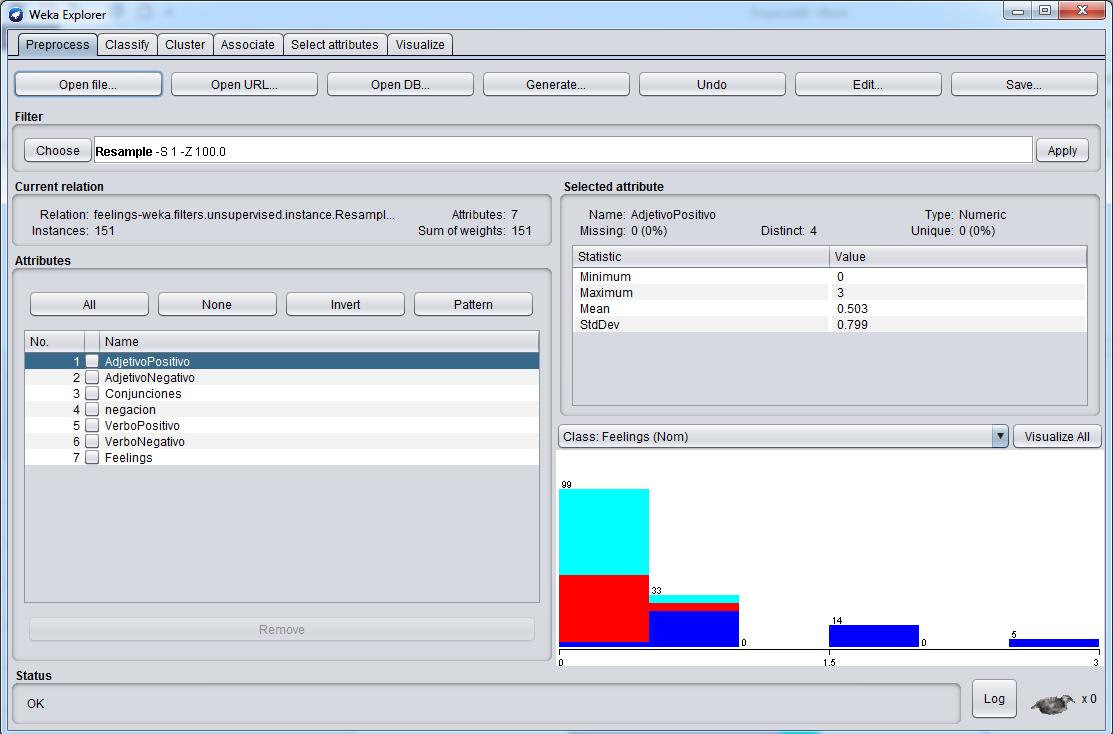
Una vez que se tiene claro los atributos y la calificación de cada uno de ellos se procede a crear el archivo arff que contiene en data set que nos permitirá aplicar algún modelo de análisis del mismo.

|  |
| --- |
| @relation feelings  @attribute AdjetivoPositivo numeric  @attribute AdjetivoNegativo numeric  @attribute Conjunciones numeric  @attribute negacion numeric  @attribute VerboPositivo numeric  @attribute VerboNegativo numeric  @attribute Feelings {positivo, negativo, neutro}  @data  1,2,2,0,0,1,negativo  1,0,0,0,0,0,positivo  0,1,1,0,0,0,negativo  1,3,1,0,0,2,negativo  0,3,1,0,0,1,negativo  .  .  .  . |

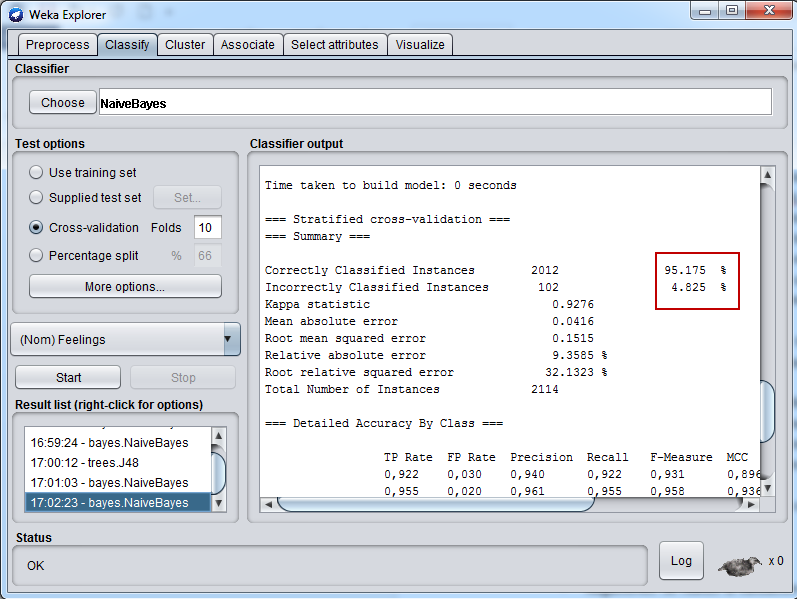
* 1. **Creación del modelo y evaluación del mismo**

Para evaluar el modelo y el dataset se utilizará la herramienta WEKA

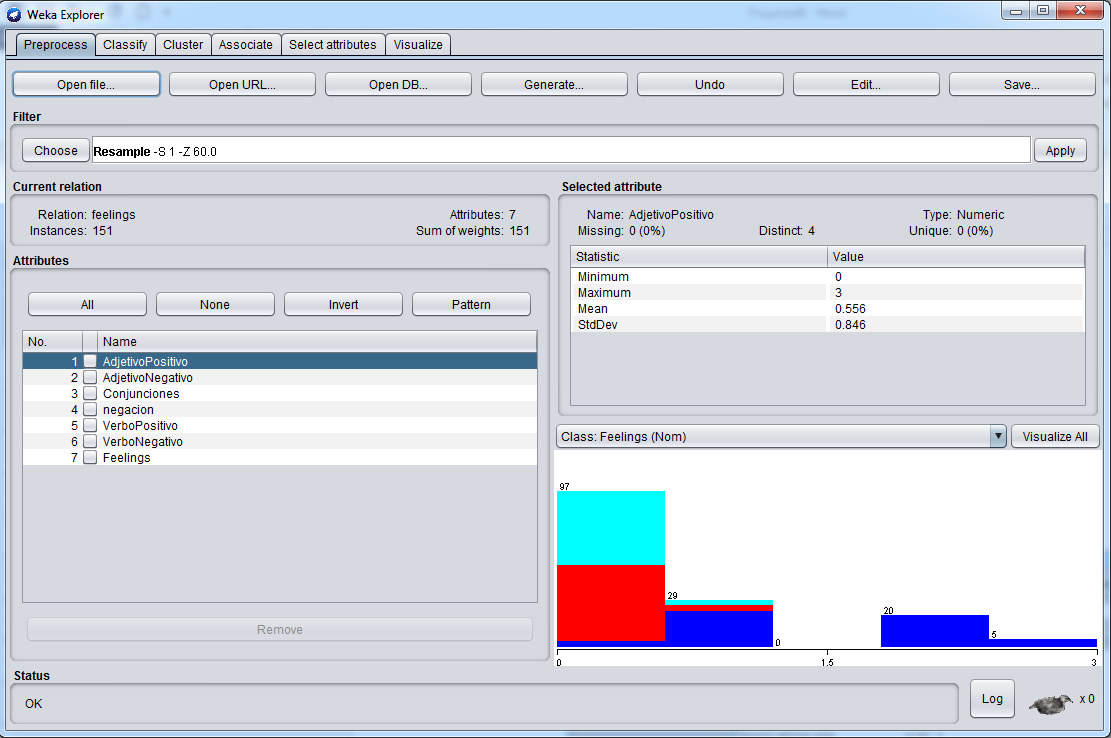
* + 1. **Algoritmo de Naive Bayes**

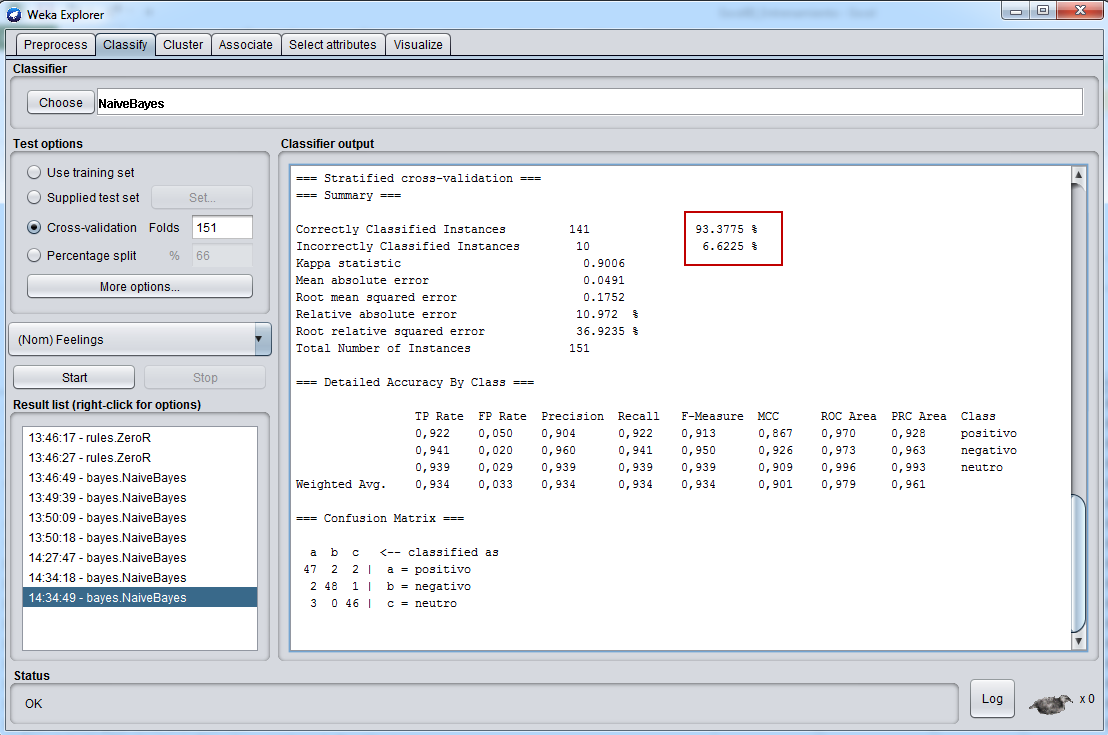


**CROSS – VALIDATION**

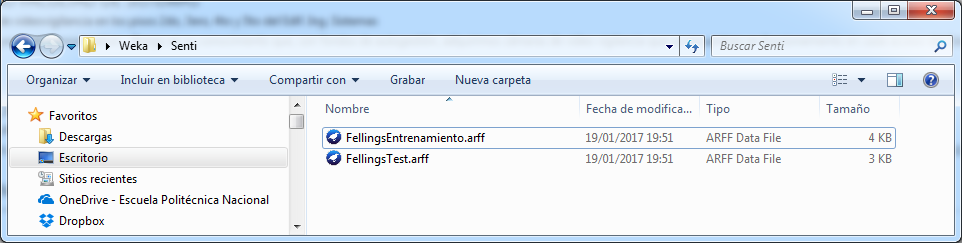


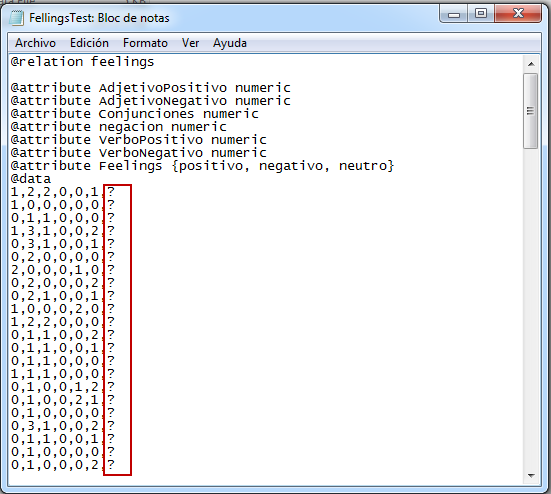
Ahora se define el data set de entrenamiento con un 60% de la muestra.



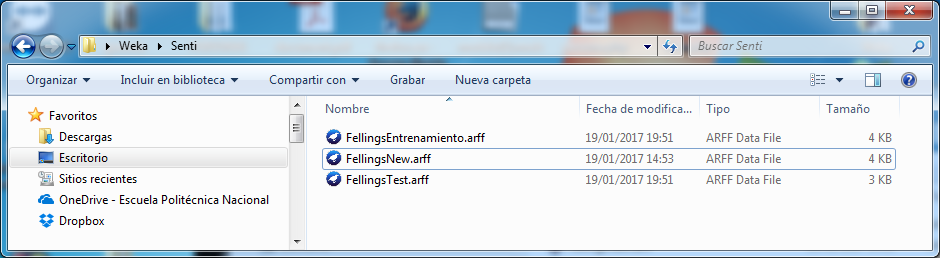


Generamos el dataset de pruebas

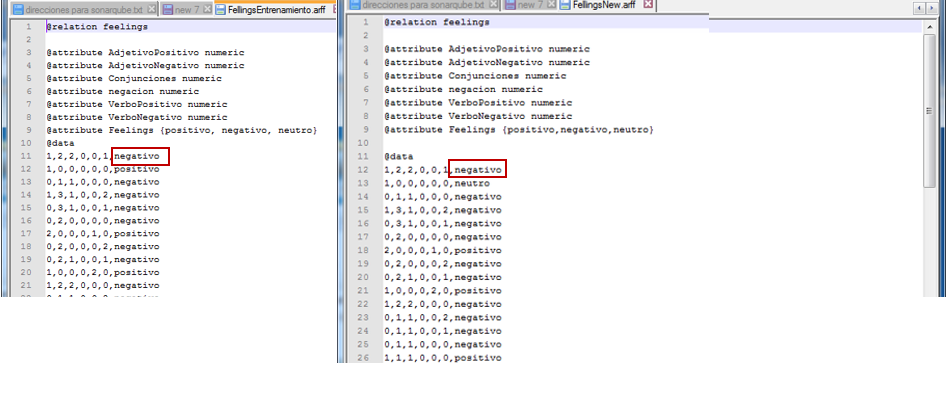




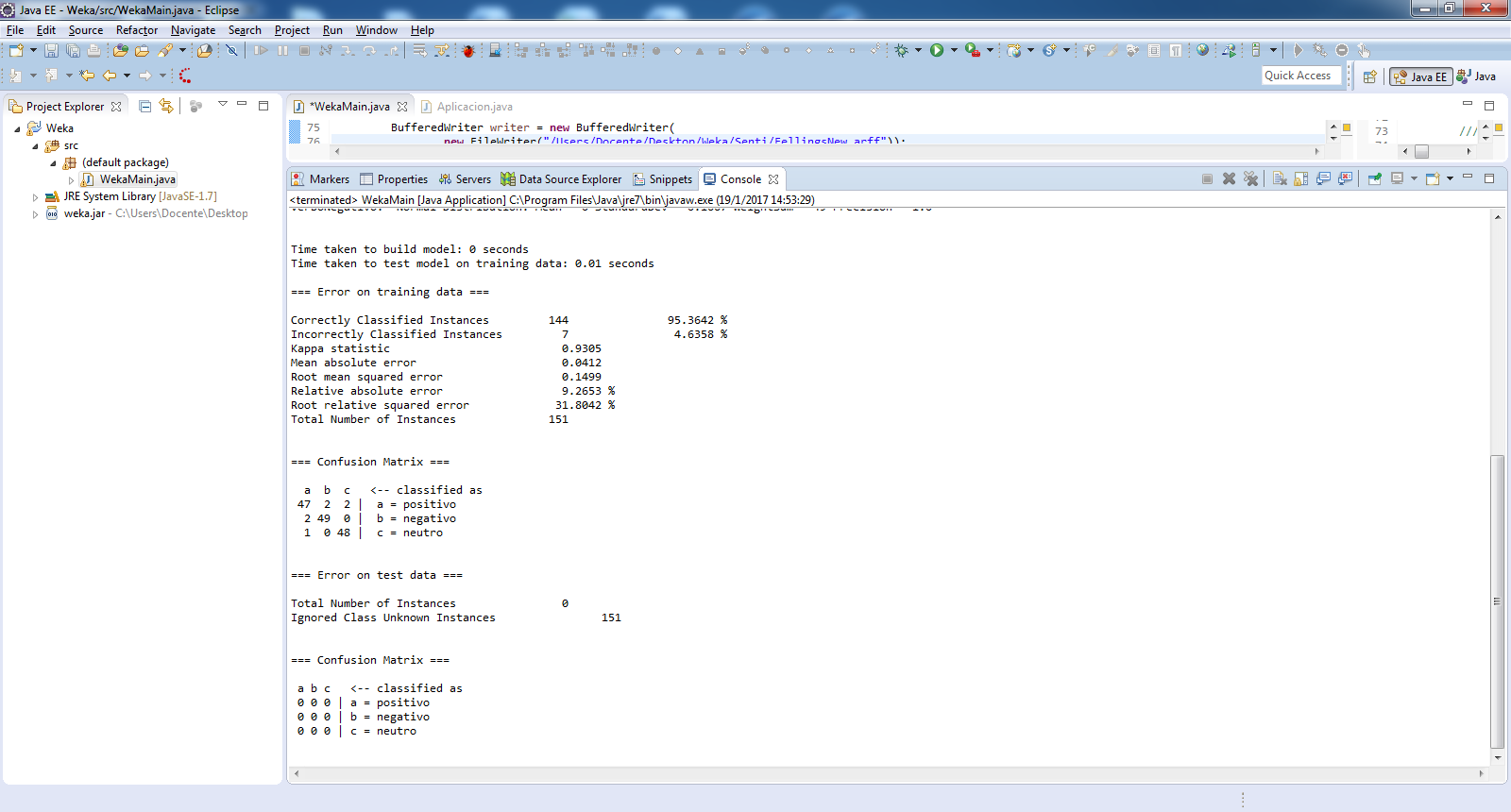
Con la ayuda del programa en Java generamos el nuevo archivo



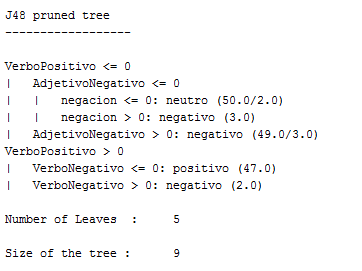
Dentro del nuevo archivo podemos verificar que los valores de la clase fueron asignados correctamente de acuerdo a la precisión calculada

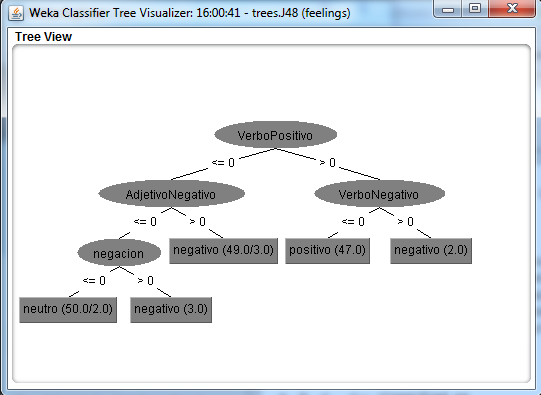


Los resultados obtenidos con el archivo Test son los siguientes:

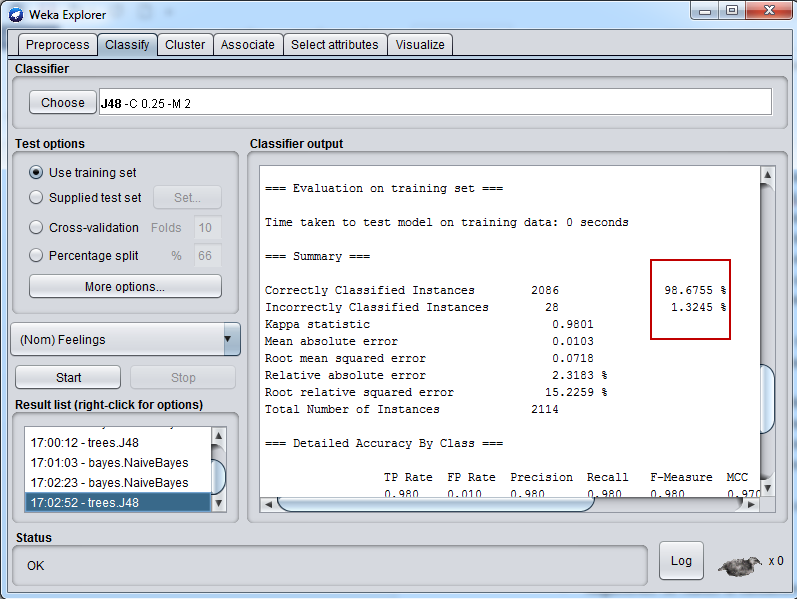


* + 1. **Algoritmo de J48 Pruned Tree**





**CROSS – VALIDATION**



1. **PREGUNTAS**
   1. **¿Cuándo se alcanza la mejor precisión?**

La mejor precisión se alcanzará cuando el modelo haya sido entrenado para recibir datos atípicos y no atípicos. Cuando mejor diseñado este el modelo la precisión será mejor.

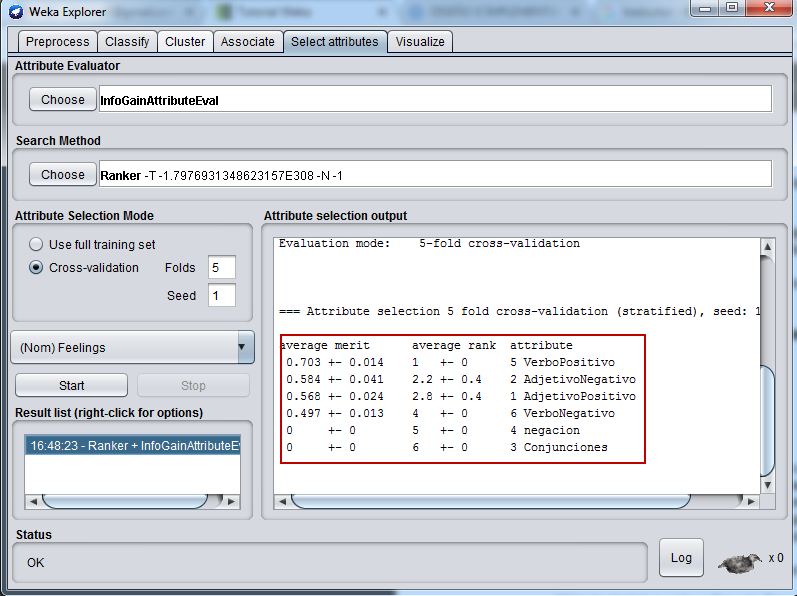
Para ello debemos enfocar nuestro esfuerzo en la definición de los atributos, pues son ellos los que nos permitirán lograr una mayor predicción de la clase.

* 1. **¿Es importante el número de atributos (features) en el clasificador?**

Si es importante, ya que si el número de atributos es excesivo, esto puede hacer que el modelo sea demasiado complejo y se produzca overfitting.

Cuantos menos atributos tenga el clasificador, más fácil de interpretar para el modelo.

Para el caso de estudio se analizó con la herramienta, que atributos son los que generan valor dentro de nuestro modelo, como se puede ver en la imagen los atributos definidos para conjunciones y negaciones no generan valor dentro del modelo.



* 1. **¿Es importante el número de instancias?. ¿En qué casos?**

Bayes Ingenuo se utiliza normalmente cuando está disponible una gran cantidad de datos, y se puede comprobar con los modelos más costosos de minería de datos.

Además se debe considerar en teneruna relación de 10 a 1 del número de casos de entrenamiento respecto al número de variables de entrada.

Según lo establece “the rule of thumb” – la regla de oro para generalización.

“Mientras más grande es dVC para una familia de hipótesis, más grande debe ser el tamaño del conjunto de ejemplos de entrenamiento. Se puede aplicar la siguiente regla de oro

**N ≥ 10dVC**

* 1. **¿Es importante considerar diferentes pesos para cada atributo? ¿por qué?**

Si, ya que a través de diferentes pesos que se les dé a los atributos, se utiliza las probabilidades para determinar a qué clase ésta instancia pertenecería.

* 1. **¿Está su modelo sobreajustado “overfitted”?**

El sobreajuste es el efecto de sobreentrenar un algoritmo de aprendizaje con unos ciertos datos para los que se conoce el resultado deseado. El algoritmo de aprendizaje debe alcanzar un estado en el que será capaz de predecir el resultado en otros casos a partir de lo aprendido con los datos de entrenamiento, generalizando para poder resolver situaciones distintas a las acaecidas durante el entrenamiento. Sin embargo, cuando un sistema se entrena demasiado (se sobreentrena) o se entrena con datos extraños, el algoritmo de aprendizaje puede quedar ajustado a unas características muy específicas de los datos de entrenamiento que no tienen relación causal con la función objetivo.

Para evitar el sobre ajuste se debe procurar buscar un equilibrio entre la cantidad de datos que tenemos y la complejidad del modelo.

Para nuestro modelo en particular, tenemos atributos que son necesarios para evaluar los sentimientos contenidos dentro de una frase. Establecimos únicamente 6 atributos para un conjunto de 2114 instancias.

* 1. **¿Los atributos continuos son mejores o peores en el clasificador Naive Bayes?**

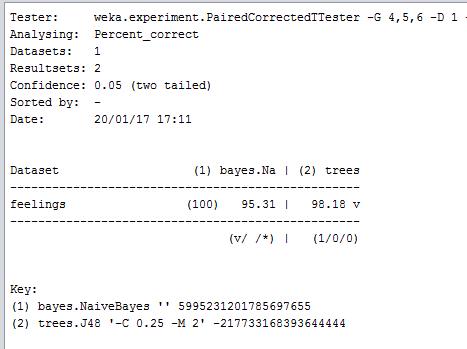
Muchos problemas del mundo real involucran cantidades continuas, tales como altura, masa, temperatura y dinero; de hecho, la mayor parte de la estadística trata con variables aleatorias cuyos dominios son continuos, Por definición, las variables continuas tienen un número infinito de valores posibles, así que es imposible especificar explícitamente las probabilidades condicionadas para cada valor. Una posible forma para manipular variables continuas es obviarlas utilizando la discretización. [3]

Por lo tanto para el clasificador de Bayes Naive no puede utilizar atributos continuos como entrada así como no pueden predecir valores continuos.

* 1. **Comparar los diferentes algoritmos con su conjunto de datos y determinar cuál de ellos es el que mejor**

Utilizando la herramienta Weka, la sección de explorer, pudimos comparar estadísticamente los resultados obtenidos aplicando:

* Naive Bayes
* J48 (arboles de decisión)



De los resultados obtenidos con la ayuda de la herramienta y con el dataset proporcionado podemos concluir que la diferencia estadísticamente utilizando el árbol de decisiones es mejor.

* 1. **¿Es mejor utilizar validación cruzada (cross-validation) o un test dataset para realizar la evaluación del clasificador? ¿Por qué?**

La validación cruzada o cross-validation se utiliza en entornos donde el objetivo principal es la predicción y se quiere estimar cómo de preciso es un modelo que se llevará a cabo a la práctica. La idea básica es estimar la calidad de cada hipótesis en la predicción de datos no observados.

Básicamente, la validación cruzada nos sirve para validar el modelo de aprendizaje.

Técnica utilizada para evaluar los resultados de un análisis estadístico y garantizar que son independientes de la partición entre datos de entrenamiento y prueba.

1. **CONCLUSIONES**

* La minería de datos es un área muy importante dentro de la generación de nuevos conocimientos, ya que nos permitió extraer información oculta en los datos para luego realizar un análisis y establecer las emociones de las personas frente al caso de estudio en este caso películas proyectadas.
* La herramienta WEKA permite realizar la clasificación de los modelos de aprendizaje utilizando algunos algoritmos, como fue el caso de árboles de decisión y Naive Bayes.
* A través de la herramienta WEKA se pudo establecer que para el set de datos utilizados no hubo una diferencia relevante que permita determinar cuál de los dos algoritmos de clasificación ofrece un mejor resultado.
* Se debe tomar en cuenta que para obtener mejor precisión se debe mejorar el modelo en relación con sus atributos mas no eliminando instancias que generan ruido o que son datos atípicos, ya que los datos atípicos con algo más de información pueden mostrar información que al momento se desconoce.
* A medida que el modelo es complejo se debe analizar el número de atributos que son necesarios, porque definir pocos o demasiados atributos puede generar un modelo sobre ajustado.
* Para nuestro modelo, los atributos negación y conjunciones de acuerdo al ranking presentado por weka no tenían ningún impacto o relevancia para el clasificador.

1. **REFERENCIAS**

|  |  |
| --- | --- |
| [1] | K. Hong, «bogotobogo,» [En línea]. Available: http://www.bogotobogo.com/Algorithms/Machine\_Learning\_NLP\_Sentiment\_Analysis\_1.php. [Último acceso: 19 01 2017]. |
| [2] | microsoft, «Métodos de discretización (minería de datos),» [En línea]. Available: https://msdn.microsoft.com/es-es/library/ms174512.aspx. [Último acceso: 19 01 2017]. |
| [3] | P. N. Stuart Russell, Inteligencia Artificial, un enfoque moderno, Madrid: Prentice hall, 2004. |

[4] Mena Elisa, "Diapositivas10 –BI 2.0", 2017.

[5] <http://ai.stanford.edu/~amaas/data/sentiment/aclImdb_v1.tar.gz>

[6] Abu-Mostafa, Y. S., Magdon-Ismail, M., y Lin, H.-T. (2012). Learning from data. AMLBook