Clasificación de Documentos usando Naive Bayes Multinomial y Representaciones Distribucionales

Juan Manuel Cabrera Jiménez[1] and Fabricio O. Pérez Pérez[1]

Instituto Nacional de Astrofísica Óptica y Electrónica, Puebla, México

1. Descripción del problema

En los últimos años, la gran popularización de los buscadores web ha permitido tener acceso a grandes cantidades de información. Esto ha resultado particularmente ventajoso en el ámbito educativo y de investigación. Sin embargo, si asumimos que un buscador web recibe de parte de un usuario consultas de texto aceptablemente claras, es evidente que en muchas ocasiones los resultados de esas búsquedas no serán las esperadas. En particular, se tendrá información redundante, dispersa o de mala calidad. Es así como surge la necesidad de encontrar mejores formas de filtrar información irrelevante, y al mismo tiempo clasificar aquella información que sí resulte de interés de acuerdo a los criterios que haya dictado el usuario.

2. Objetivos del proyecto

2.1. Marco Teórico

El problema de clasificar contenidos web es un caso particular de un problema más general: la clasificación de textos. Las tareas involucradas en esta área de investigación han obtenido un lugar en los sistemas de información, debido a la gran cantidad de documentos en formato digital que se requiere organizar. Por ello, resulta factible proponer una solución al problema que se ha planteado en la sección anterior. Para lograr este fin, se dará una breve explicación de la terminología utilizada.

Algunas formas de representación de datos

- Bolsa de palabras. El texto es una colección desordenada de palabras, sin importar la gramática. Se conoce la frecuencia de cada término y se asume que son independientes entre sí dada la clase del documento [**?**].
- Distribucional DOR. El texto se percibe como una colección de documentos, los cuales se definen en función de la importancia de algunos términos y la irrelevancia de otros términos [**?**].
- Distribucional TCOR. El texto se percibe como una colección de términos, donde cada uno de ellos se define a partir de una distribución sobre los términos que co-ocurren con él con frecuencia [**?**].

Clasificación de textos Se han empleado diversos modelos de clasificación, entre los cuales figuran los clasificadores Naive-Bayes. La naturaleza de estos modelos exige que los términos sean independientes dada la clase de documento. Aunque tal independencia no siempre es cierta, resulta que estos clasificadores son relativamente sencillos de implementar y son capaces de producir soluciones muy buenas en comparación con aquellas que se han obtenido a través de métodos más sofisticados (de mayor complejidad algorítmica).

En este trabajo se considerarán dos clasificadores de la familia Naive-Bayes: bayesiano simple (binomial) y bayesiano multinomial.

- Clasificador bayesiano simple (binomial). Considera de forma binaria la probabilidad de aparición de cada término dada la clase de documento, es decir, el término aparece o no en la clase.
- Clasificador bayesiano multinomial. Considera el número de apariciones de cada término para evaluar la contribución de su probabilidad condicional dada la clase del documento.

Alternativamente, si el modelo no puede omitir dependencias entre términos, otra solución es considerar modelos TAN (Tree Augmented Naive-Bayes) o BAN (Bayesian Augmented Naive-Bayes).

Medidas de Rendimiento En la evaluación del desempeño de los clasificadores se utilizan diversas medidas. Sean:

- a - número de documentos bien clasificados (positivo).
- b - número de documentos incorrectamente clasificados (falso positivo).
- c - número de documentos incorrectamente rechazados (falso negativo).

Las que serán empleadas en este trabajo son el recuerdo, la precisión, macro-promedio y micro-promedio:

$$Recuerdo = \frac{a}{a+c} \tag{1}$$

$$Precision = \frac{a}{a+b} \tag{2}$$

- Macro-promedio. Primero se calculan las medidas de rendimiento (recuerdo y precisión) por clase. Luego se promedian estas medidas para obtener las medias globales. Da el mismo peso a cada clase.
- Micro-promedio. Primero se calculan los totales a, b y c para todas las clases. Después se usan estos totales para calcular las medidas de rendimiento (recuerdo y precisión). Da el mismo peso a cada documento.

2.2. Naive Bayes Multinomial

Existen 2 modelos usados comúnmente para la clasificación de textos: el modelo de eventos multivariado de Bernouli y el modelo de eventos multinomial [**?**]. Al modelo de eventos multivariado frecuentemente se le conoce como Naive Bayes Multinomial(NBM), que generalmente supera al Multivariado [**?**] y también se han comparado con otros modelos más especializados. Sin embargo, es inferior a las máquinas vectoriales de soporte (support vector machines) en términos de exactitud cuando se trata de clasificación de textos.

Se basa en la aplicación de la Regla de Bayes para predecir la probabilidad condicional de que un documento pertenezca a una clase ($P(c_{ij}|d_j)$) a partir de la probabilidad de los documentos dada la clase $P(d_j|c_i)$ y la probabilidad a priori de la clase en el conjunto de entrenamiento $P(c_i)$.

$$Pr(c|t_i) = \frac{Pr(c)Pr(t_i|c)}{Pr(t_i)} \tag{3}$$

Donde la clase Pr(c) puede ser estimada dividiendo el numero de documentos que pertenecen a las clase c por el total de documentos. $Pr(t_i|c)$ es la probabilidad de obtener un término t_i en la clase c y se calcula de la siguiente forma:

$$Pr(t_i|c) = \left(\sum_n f_{ni}\right) \prod_n \frac{Pr(w_n|c)^{f_{ni}}}{f_{ni}} \tag{4}$$

Donde f_{ni} es el numero de palabras n en el conjunto de prueba t_i y $Pr(w_n|c)$ es la probabilidad de una palabra n dada una clase c, $Pr(w_n|c)$ es estimada del conjunto de entrenamiento como :

$$Pr(w_n|c) = \frac{1 + F_{nc}}{N + \sum_{x=1}^{N} F_{xc}} \tag{5}$$

Donde F_{xc} es el numero de palabras x en todo el conjunto de entrenamiento que pertenecen a la clase c. Al normalizar la ecuación 1 debido al problema de la frecuencia 0 resulta:

$$Pr(t_i) = \sum_{k=1}^{|C|} Pr(k)Pr(t_i|k) \tag{6}$$

Debido a que es computacionalmente muy caro realizar las operaciones $\sum_n f_{ni}!$ y $\prod_n f_{ni}!$ en la ecuación 2, podemos borrarlas sin afectar los resultados ya que no dependen de la clase. Como resultado obtenemos la ecuación final:

$$Pr(t_i|c) = \alpha \prod_n Pr(w_n|c)^{f_{ni}} \tag{7}$$

2.3. Hipótesis de Solución

Usando la representación distribucional DOR se tendrá un mejor desempeño en la clasificación de las páginas web, en términos de precisión y recuerdo, usando clasificadores bayesianos multinomiales.

2.4. Objetivo General

Observar las ventajas de la representación distribucional DOR frente a otras formas de representación en tareas de clasificación de documentos. Observar las ventajas de los clasificadores Bayesianos Multinomiales frente a los Bayesianos simples.

2.5. Objetivos Específicos

- Establecer los procedimientos que se usarán para formular y evaluar los clasificadores.
- Determinar las formas de representación con las que se van a comparar las representaciones distribucionales.
- Desarrollar el clasificador bayesiano multinomial.
- Realizar pruebas combinando las representaciones con los clasificadores:

Tipo de Representación	**Clasificador**
Bolsa de palabras	Naive Bayes
DOR	Naive Bayes
Bolsa de palabras	Naive Bayes Multinomial
DOR	Naive Bayes Multinomial

- De acuerdo a alguna de las representaciones elegida en turno, obtener los resultados de cada uno de los clasificadores.
- Analizar y comparar los resultados de cada clasificador de acuerdo a la precisión, recuerdo, macro-promedio y micro-promedio.

3. Técnica(s) del curso que se aplican

En este proyecto se pretende utilizar el clasificador bayesiano simple visto en clase. Se utilizará también el clasificador bayesiano multinomial, una generalización del anterior.

4. Procedimiento

4.1. Conjunto de datos

Para los experimentos usaremos el corpus WebKB de páginas web de 4 universidades, los cuales ya están etiquetados en 4 clases (course, project, faculty, student). Primero experimentaremos con la representación de bolsa de palabras y los dos clasificadores bayesianos (NB y NBM), observaremos el comportamiento y los resultados. Después experimentaremos con la representación distribucional DOR y los mismos dos clasificadores bayesianos (NB y NBM), compararemos los resultados obtenidos con los anteriores en términos de recuerdo, precisión,

Categoría	Conjunto
course	929
project	504
faculty	1124
student	1641
Total	4198

Cuadro 1. Conjunto de datos WebKB con 4 clases

macro-average y micro-average. Para ambos casos se usará el pesado tf-idf (term frecuency - inverse document frequency).

Para realizar una comparación de este método de representación con el NBM nos basamos en los resultados obtenidos de una tesis de maestría [**?**], con la cual se obtuvieron los resultados mostrados en la tabla 1.

Figura 1. Resultados de [**?**] usando reśumenes automáticos

	Esquema del Uso de Resúmenes			
Tamaño de Resumen	Doc-Doc	Res-Res	Doc-Res	Res-Doc
10 %	**.607**	.488	.554	.496
20 %	**.607**	.52	.481	.575
30 %	**.607**	.522	.459	**.607**
40 %	**.607**	.542	.459	.606
50 %	**.607**	.556	.456	**.607**
60 %	**.607**	.567	.461	.606
70 %	**.607**	.573	.466	.599
80 %	**.607**	.587	.471	.603
90 %	**.607**	.593	.464	.601

4.2. Arquitectura

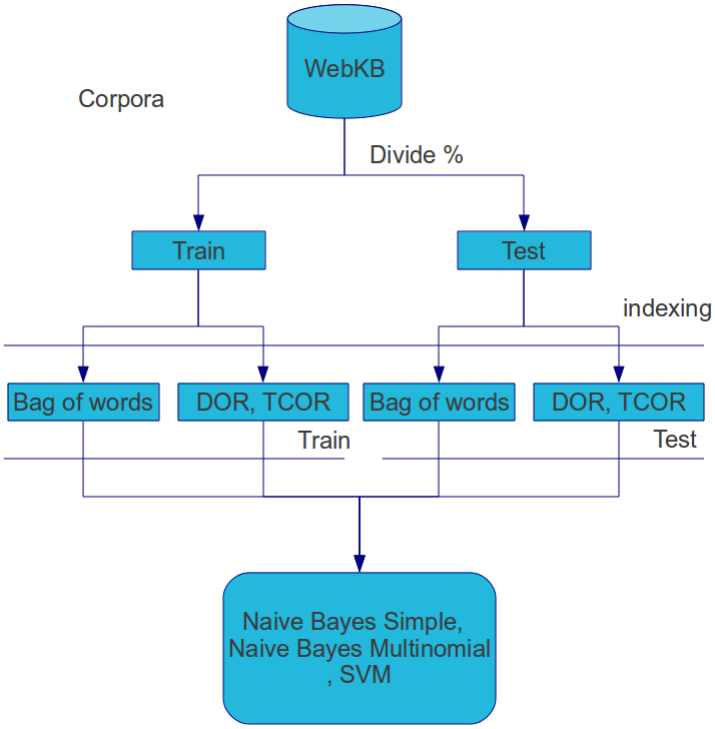
La primera parte del procesamiento incluye dividir los datos corpora WebKB en dos grupos, uno de entrenamiento y uno de pruebas, la razón es que los pesos de las representaciones utilizadas (*tf-idf*) emplean información de la colección, por lo que si utilizamos toda la colección tendríamos información adicional referente al conjunto de pruebas, situación que alteraría los resultados. Los aspectos que intervienen en esta decisión son:

- Si se utiliza toda la colección, el *tf-idf* hace uso de la información de la colección de documentos, cosa que resultaría aparentemente ventajosa (pero poco realista), ya que se conocería información del conjunto de pruebas.

El $idf = log\left(\frac{|M|}{N}\right)$, donde N es el número de documentos en los que aparece un término dado y M es el conjunto de términos distintos en todo el dominio del problema (vocabulario). En este caso se contemplarían también los documentos de prueba.

- Sería un problema intentar usar toda la colección, pues DOR representa los términos en función de los documentos que co-ocurren con él.

Después de dividir los datos en dos subconjuntos, el siguiente paso es realizar la representación de los documentos, es decir, se crean matrices tanto para la representación de bolsas de palabras como para la representación distribucional DOR. Una vez hecho esto, se procede a realizar la clasificación.



4.3. Herramientas

Para la generación de las representaciones distribucionales DOR, se utilizará Matlab en su versión 2010, además de Text to Matrix Generator(TMG), la cual es una herramienta (escrita en Matlab y para Matlab) de investigación y docencia para la generación de TDM (*matrices de términos-documentos*, por sus siglas en inglés) de colecciones de texto y para modificaciones de estas matrices. Para la clasificación, los algoritmos de Naive Bayes y Naive Bayes Multinomial se desarrollarán en el lenguaje de programación Java.

4.4. Resultados y Análisis

Experimento 1: Clasificadores Uno de los objetivos de este proyecto es la comparación de dos clasificadores: Naive Bayes (NB) y Naive Bayes Multinomial (NBM). Para esto se implementó el algoritmo Naive Bayes Multinomial en el lenguaje de programación java, utilizando las librerías del software WEKA. Además se comparó el resultado con el algoritmo NBM que ya esá implementado en WEKA (NBMW). Se utilizaron tres conjuntos de datos, dos pequeños (Iris y Wine) y uno grande (WebKB), realizando la comparación con una confianza del 95 %, como se muestran en la Tabla .

Conjunto de datos	NB	NBM	NBMW
WebKB	77 %	79.59 % v	79.60 % v
Wine	97 %	80.53 % *	85.02 % *
Iris	94.89 %	95.11 %	95.11 %

Cuadro 2. Pruebas para los clasificadores. En el caso de WebKB sólo se utilizó el 40 %

Se puede observar que para el conjunto más grande se obtuvieron mejores (v) resultados que el baseline (en este caso NB). Sin embargo, para los otros dos conjuntos de datos (*Iris* y *Wine)* se obtuvieron peores resultados: ambos tuvieron la misma confianza. Por los resultados obtenidos anteriormente, se realizó otro experimento con conjuntos de datos más grandes que Iris y Wine. Se seleccionaron los conjuntos de datos *glass* y *spambase*, pero no se obtuvieron mejores resultados que el baseline NB en cuanto a porcentaje de clasificación correcta, como se muestra en la Tabla 3.

Conjunto de datos	NB	NBM	NBMW
Glass	47.5 %	51.43 %	51.43 %
Spambase	79.56 %	75.23 % *	79.15 %

Cuadro 3. Conjunto de pruebas para los clasificadores

Experimento 2: Representación DOR Con el objetivo de comprobar si el desempeño de la representación DOR usando el clasificador Naive Bayes Multinomial podrá mejorar la clasificación de documentos (páginas web), en comparación con la representación de bolsa de palabras usando TF-IDF únicamente, se usó el conjunto de datos WebKB. Para la representación de los documentos se dividió el conjunto en dos (entrenamiento y pruebas) mostrados en la Tabla .

Al conjunto de entrenamiento se le realizó stemming con el propósito de eliminar confusiones semánticas que se puedan presentar, además de reducir el vocabulario y por lo tanto la dimensionalidad del conjunto de datos. El total de palabras diferentes del vocabulario es de 36000. Además de utilizar stemming, sólo se utilizaron palabras con una longitud mínima de 2 y con una frecuencia global de al menos 5 (que por lo menos aparezcan 5 veces en todo el conjunto de entrenamiento). Una vez aplicado este filtro, se obtuvieron 6294 atributos (palabras) con los que se realizaron los siguientes experimentos.

De la misma forma que al conjunto de entrenamiento, al conjunto de prueba se le aplicó el mismo filtrado (stemming, frecuencia mayor a 5 y longitud mínima de palabra de 2). Después de realizar el pre-procesamiento de los dos conjuntos de datos se procedió a representar a los documentos usando DOR.

Al conjunto de entrenamiento se le representó usando su frecuencia, y a su vez, con esas frecuencias se representó a DOR. Sin embargo, al conjunto de pruebas no se le dio el mismo tratamiento, debido a que están separados en dos archivos diferentes y para el de prueba únicamente se calculó de forma binaria, es decir, se encuentra o no el término, mientras que para la representación DOR se tomó como la sumatoria de los vectores (cada uno representa un término) que están presentes en el documento de prueba (como se muestra en la Figura 2). En el caso de que el documento de prueba únicamente contenga el término B, se sumará el vector de ese término.

Figura 2. Ejemplo de la representación DOR

Término	D1	D2	D3	D4
A	.5	.5	0	.5
B	.7	0	.7	0
C	.5	.5	.5	.5
D	0	0	.7	.7
F	0	1	0	0

Una vez realizada la representación para los documentos de prueba, procedemos a convertirlos al formato que utiliza Weka (ARFF), para probar los datos obtenidos con los dos clasificadores.

Experimento 3: Representación DOR y NB El clasificador Naive Bayes junto con la representación DOR obtuvieron un buen desempeño, aún cuando se tiene un desbalance en el conjunto de datos. En la tabla 4 puede verse la matriz de confusión resultante de la clasificación, con la que se obtuvo 79.89 % de eficiencia.

Course	Faculty	Project	Student		Precisión	Recuerdo	F-Measure
127	1	2	1	Course	.876	.969	.92
6	113	30	11	Faculty	.801	.706	.751
3	4	61	3	Project	.513	.859	.642
9	23	26	172	Student	.92	.748	.825

Cuadro 4. Matriz de confusion con NB

Experimento 4: Representación DOR y NBM Para el siguiente experimento ahora se utilizó el clasificador Naive Bayes Multinomial, obteniendo el 77.87 % de eficiencia.

Course	Faculty	Project	Student		Precisión	Recuerdo	F-Measure
117	2	2	10	Course	.967	.893	.929
1	118	6	35	Faculty	.698	.738	.717
1	17	42	11	Project	.737	.592	.656
2	32	7	189	Student	.771	.822	.796

Cuadro 5. Matriz de confusion con NBM

Como veremos más adelante, los clasificadores dependen mucho del conjunto de datos con los que se pruebe, por lo que los resultados pueden variar. Sin embargo, podemos observar que tenemos resultados por encima del baseline con los dos clasificadores.

Un dato importante es al considerar la palabras más relevantes de la tabla 5. Podemos observar que el número de palabras con un peso mayor a 0.8 varía, haciendo pensar que este resultado ha sido influído por el número de documentos. Sin embargo, eso no es del todo cierto, pues existen más documentos de la clase 1 que de la clase 3. También podemos observar que varias de ellas tienen la longitud mínima requerida, esto quizás se debe al truncamiento aplicado en el proceso de stemming, por lo que se procede a realizar otro experimento aumentando la longitud mínima y la frecuencia.

Experimento 5: Representación DOR con longitud mínima 4, usando NB y NBM Para comprobar que los términos de longitud 2 no son tan relevantes, realizamos la siguiente prueba donde aumentamos la longitud mínima de las palabras a 4 y con una frecuencia mínima global de 8, obteniendo los siguientes resultados: con NB 77.87 % y con NBM 78.20 %

La siguiente gráfica nos muestra la comparación de los dos clasificadores usados junto con la representación DOR con sus distintas opciones usadas, en la cual podemos observar que se ha superado al baseline que usa representación de bolsa de palabras y clasificador NB. La diferencia existente entre el primer

course	sd, condens,dsp
student	clear, mitra, stoller, assag, bobbi, fujimoto, duncan, lick, amer, diwan, hilbert, abram, timer, pku, flag, heather, sen, fp, rashid, shaffer, csp, sin, christma, shake, ira, horn, mcauliff
project	multipol, suspens, dump, scalapack, immers, envi, insensit, shtml, adjac, arl, pim, condor, exodu
faculty	reid, fell, gale, bishop,rinard, landau, can, lyon, feiner, kaiser, conwai, andersen, stark, harm, clinton, sussman

Cuadro 6. Palabras relevantes

Figura 3. Matriz de confusión con NB y una LNW 4 y Frec. 8

Course	Faculty	Project	Student		Precision	Recuerdo	F-Measure
126	0	0	5	Course	.906	.962	.933
4	111	31	14	Faculty	.787	.694	.738
2	3	60	6	Project	.488	.845	.619
7	27	32	164	Student	.868	.713	.783

Figura 4. Matriz de confusión con NBM y una LNW 4 y Frec. 8

Course	Faculty	Project	Student		Precision	Recuerdo	F-Measure
116	3	1	11	Course	.959	.885	.921
1	116	3	40	Faculty	.703	.725	.714
2	18	38	13	Project	.776	.535	.633
2	28	7	193	Student	.751	.839	.793

experimento y el segundo no es significante.

En la gráfica 5 podemos observar la comparación del tiempo de creación del modelo de cada clasificador. Vemos que el NBM tiene un aprendizaje más rápido comparado con el NB.

Experimento 6: Conjunto completo de WebKB Como experimento final se quiso ver cómo se comporta la representación DOR con todas las clases del conjunto de datos de WebKB (tabla 7). Se trabajó con una longitud mínima de palabras de 5 y una frecuencia global de 10, opciones elegidas de esta manera debido a la limitación de hardware y tiempo.

Figura 5. Comparación de Tiempos

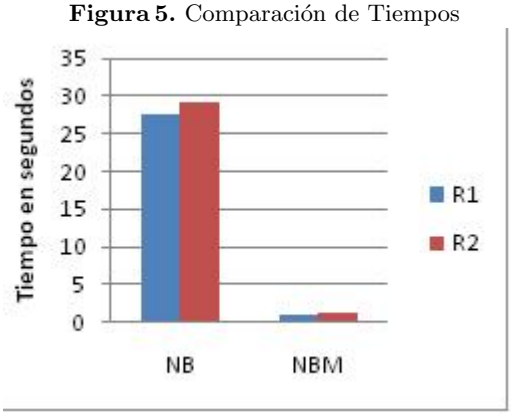
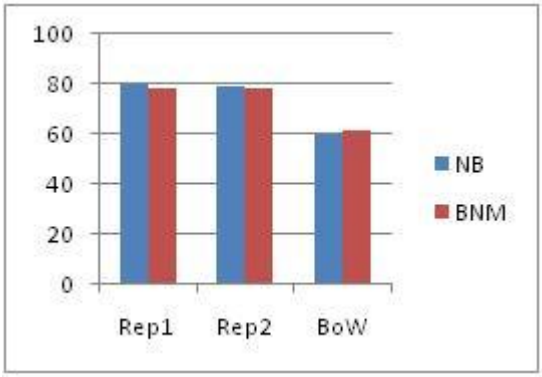


Figura 6. Porcentaje de clasificación correcta



Se obtuvieron resultados alentadores, ya que se superó al baseline en un 7 % (como se muestra en la tabla 8), por lo que podemos notar que *staff* obtuvo un valor de F-measure menor, pero también esto se debe a la cantidad de datos que se tenían para realizar las pruebas. En general, tanto *staff* como *department* están muy desbalanceadas con respecto a las otras clases. De estos resultados también podemos observar que los documentos de la clase *cursos* son los que están mejor redactados, o por lo menos más identificables en comparación con los de *estudiante*, que son un caso general de *course*, *faculty* y *project*.

Cuando se cambió al clasificador NBM se mejoró un poco más la clasificación (tabla **??**) llegando a un 71.87 % de eficiencia. Sin embargo, todas las instancias de la clase *staff* y *department* fueron clasificadas incorrectamente, lo que puede deberse a la forma en que el NBM trabaja, puesto que al contar las frecuencias de las palabras, el clasificador tiene muy poca evidencia de estas clases con respecto a las otras.

Clase	Entrenamiento	Pruebas
course	768	131
student	1337	267
project	417	83
faculty	932	187
department	152	30
Staff	11	23
Total:	3717	743

Cuadro 7. Conjunto de datos WebKB con todas las clases

Course	Faculty	Project	Student	Staff	Department		Precisión	Recuerdo	F-Measure
138	2	7	0	0	6	Course	.879	.902	.89
4	110	35	15	7	16	Faculty	.815	.588	.683
1	4	30	4	4	10	Project	.476	.723	.574
14	19	20	150	41	23	Student	.847	.562	.676
0	0	4	6	13	0	Staff	.2	.565	.295
0	0	0	2	0	28	Department	.337	.933	.496

Cuadro 8. Matriz de confusión con NB

Course	Faculty	Project	Student	Staff	Department		Precisión	Recuerdo	F-Measure
138	1	5	9	0	0	Course	.939	.902	.92
4	130	9	44	0	0	Faculty	.66	.695	.677
0	19	41	23	0	0	Project	.707	.494	.582
5	35	2	225	0	0	Student	.66	.843	.74
0	0	9	1	13	0	Staff	0	0	0
0	3	0	27	0	0	Department	0	0	0

Cuadro 9. Matriz de confusión con NBM

5. Conclusiones

Entre las conclusiones a las que se ha llegado mediante la realización de este trabajo, es que la clasificación tiene una fuerte dependencia de tres aspectos: el conjunto de datos (pre-procesamiento), la representación de los documentos y el clasificador utilizado.

- **Conjunto de Datos (pre-procesamiento)**
  Factor muy influyente en nuestro trabajo para la obtención de de resultados, ya que las páginas web tienen una fuerte relación con la posición en la que se encuentra el texto y las etiquetas que las rodean, como son {h1, h2, title, ...}. Además, es un poco complicado separar los meta-tags del texto y ocasionalmente se incluyó accidentalmente texto de los tags.

- **Representación de los datos**
  Las forma en que los datos se representan junto con el clasificador mejoran o empeoran los resultados, por lo que la elección de una adecuada forma

de representar a los documentos con un clasificador ya es un problema. En particular, al usar DOR, la semántica expresada por el contexto ayuda a mejorar la clasificación. También puede notarse que el peso que asigna el documento al término ayuda en gran medida a capturar el contexto (caracterizando mejor a los términos), ya que en DOR el significado de un término está dado como la suma de los contextos en los que ocurre, que en este caso los contextos son definidos como documentos (páginas).

- **Clasificación**
  El clasificador NBM demostró un buen desempeño al tomar en cuenta las frecuencias de apariciones de cada atributo. Sin embargo, es importante hacer notar que en todos los corpus utilizados NBM no pudo mejorar al NB, lo cual puede deberse a la forma de los datos, suposición que se pretende comprobar como trabajo futuro.

Referencias

1. K. Aas and L. Eikvil. Text categorization: A survey. 1999.
2. F. S. Alberto Lavelli. Distributional term representations: An experimental comparison. *Italian Workshop on Advanced Database Systems*, 2004.
3. E. A. Hernández. Método semisupervisado para clasificación de documentos usando resumenes automáticos. Master's thesis, INAOE, 2010.
4. A. N. K. McCallum. A comparison of event models for naive bayes text classification. *American Association for Artificial Intelligence Workshop on Learning for Text Categorization*, 1998.
5. K.-M. Schneider. Techniques for improving the performance of naive bayes for text classification. *Springer-Verlag Berlin Heidelberg 2005*, 2005.