Evaluación de técnicas de Aprendizaje Activo para codificación CIE-9-MC de informes de alta hospitalaria

David Lojo[1], David E. Losada[2], Álvaro Barreiro[3]

[1] Servicio de Informática. Complexo Hospitalario Universitario de Santiago
Santiago de Compostela, Spain
[2] Grupo de Sistemas Inteligentes, Dep. de Electrónica y Computación
Universidade de Santiago de Compostela, Spain
[3] IRLab. Dep. de Computación
Universidade da Coruña, Spain

Abstract.
El Aprendizaje Activo es una técnica según la cual, a partir de un conjunto de documentos sin etiquetar, se ordenan y seleccionan los documentos para ser etiquetados de modo que el nuevo conjunto de entrenamiento mejore el clasificador construido. En los hospitales se genera un gran volumen de información, pero sólo se codifica una pequeña parte de los informes producidos. Es por tanto un escenario donde se necesita elegir bien lo que se etiqueta para que las herramientas automatizadas de clasificación puedan surtirse de buenos conjuntos de entrenamiento. En nuestro trabajo, vamos a utilizar técnicas de Aprendizaje Activo para elegir los informes de alta hospitalaria que se deben etiquetar con códigos CIE-9-MC y, a continuación, evaluaremos la calidad de ese proceso de selección. Los documentos se representan utilizando técnicas populares en Recuperación de Información y la calidad de los conjuntos de entrenamiento se evalúa utilizando clasificación con Máquinas de Soporte Vectorial. El dominio clínico donde trabajamos es muy complejo, con un gran número de clases, y existe desbalanceo y poca independencia entre las clases. Los resultados de experimentación demuestran que nuestra estrategia es prometedora para mejorar este tipo de sistemas.

1 Introducción

En los hospitales se genera un gran volumen de información con considerable complejidad. La capacidad de clasificación manual es limitada por lo que es imposible que todos los documentos producidos sean etiquetados. Una de las tareas de clasificación que se realizan es la codificación de los diagnósticos de los informes de alta. La codificación es un proceso que consiste en analizar la documentación del alta, y asignar los códigos de los diagnósticos de ese episodio clínico. Este proceso se realiza de forma manual por un médico codificador, con un gran coste por la complejidad del tipo de clasificación, ya que consiste en la asignación de algunos de

los más de 21.000 códigos que tiene el CIE-9-MC [1] de la Organización Mundial de la Salud (OMS).

Se trata de un problema de clasificación con múltiples clases. Un documento puede pertenecer a varias clases, y el número de clases es variable para distintos documentos. En los hospitales, los episodios que se codifican habitualmente son los ingresos hospitalarios. Otros episodios clínicos no son codificados (por ejemplo los que corresponden a episodios de consultas externas, urgencias, pruebas funcionales, etc.). Actualmente el porcentaje de episodios clínicos que se codifican con respecto al total es mínimo. Si quisiésemos codificar todos los episodios clínicos que se generan en un centro hospitalario, tendríamos que aumentar de forma considerable los recursos humanos de médicos codificadores, lo que implicaría un elevado coste económico.

Debido a estas limitaciones los episodios clínicos pasan usualmente por una clasificación generalista, simplemente para generar una contabilidad básica, sin considerar la patología tratada para cada paciente. En cambio, con la codificación CIE-9-MC completa de estos episodios podríamos medir, comparar y mejorar la calidad asistencial, agrupando a los pacientes de acuerdo a requerimientos y características comunes.

En la literatura existen propuestas de clasificación automática para dar soporte a la codificación de informes [2,3,4]. Para poder aplicar técnicas de clasificación automática a los episodios clínicos, debemos elegir bien los episodios que actúan como entrenamiento para el clasificador. Para ello proponemos aplicar técnicas de Aprendizaje Activo (AA), para seleccionar aquellos documentos con los cuales poder entrenar un clasificador con resultados eficaces. El aprendizaje activo [5,6] es una técnica empleada en el entrenamiento de clasificadores que a partir de un conjunto de documentos sin etiquetar, escoge los documentos más informativos para la construcción del clasificador, obteniendo un conjunto etiquetado con una alta capacidad discriminante.

En este artículo evaluamos AA sobre una colección real del dominio clínico que presenta alto desbalanceo entre clases y en la que el problema de clasificación es difícil. Nuestros experimentos demuestran que AA se puede utilizar en este escenario con resultados razonables.

El resto del documento está organizado de la siguiente forma. Las estrategias de aprendizaje activo para la clasificación de textos multietiqueta se describe en la sección 2. En la sección 3 se concreta el método utilizado, para terminar con los experimentos en la sección 4 y las conclusiones en la sección 5.

2 Aprendizaje activo para la clasificación de textos multietiqueta

El Aprendizaje Activo (Active Learning), consiste en seleccionar los ejemplos más informativos entre los no etiquetados con la finalidad de etiquetarlos y agregarlos al conjunto de entrenamiento. Con esta técnica intentamos reducir el coste a la hora de obtener una colección etiquetada mediante la selección de los mejores documentos para el sistema de aprendizaje.

En muchas ocasiones en un esquema de aprendizaje supervisado la colección de entrenamiento es pequeña, y es muy costoso obtener nuevos documentos etiquetados. Sin embargo, se suele disponer de una gran cantidad de documentos sin etiquetar. Esta es la situación que nos encontramos en los hospitales en donde para ciertas áreas clínicas no disponemos de una colección etiquetada de sus episodios. Además la etiquetación manual es costosa y ha de ser realizada por expertos.

Para conseguir la colección de entrenamiento aplicamos AA en un entorno multietiqueta. Dado un conjunto de clases predefinidas $C = \{C_1, \ldots, C_m\}$, y una colección de documentos a clasificar *(D)*, el objetivo es encontrar una función con la forma $\varphi : D \times C \rightarrow \{-1,+1\}$, que denominamos clasificador (-1 y +1 representan la pertenencia o no a una clase). Un documento puede no pertenecer a ninguna clase, a una clase o a varias clases. En una clasificación de textos multietiqueta usualmente se generan *m* clasificadores binarios, uno por cada clase c_j. En nuestro trabajo nos centramos en clasificadores que tienen la forma $\varphi^* : D \times C \rightarrow [-1,+1]$. Esta función permite estimar la clase a la que el clasificador cree que el documento pertenece (signo de $\varphi^*(d_i,c_j)$) y, además proporciona una estimación de la confianza del clasificador en la decisión tomada, $|\varphi^*(d_i,c_j)|$.

Las técnicas de AA para entornos con una única clase consisten básicamente en escoger primero aquellos ejemplos no etiquetados sobre los que el clasificador automático tiene menos confianza. El codificador humano de esta forma etiquetará manualmente solo aquellos documentos más informativos para el aprendizaje. En situaciones como la nuestra, con múltiples clases, cada documento tiene un valor de confianza para cada clase por lo que es necesario combinar esas puntuaciones para estimar qué documento no etiquetado es globalmente más informativo para el proceso de clasificación multietiqueta.

En la literatura [7], nos encontramos dos opciones para seleccionar los documentos que vamos a incorporar al conjunto de entrenamiento. Por un lado, generar *m* rankings independientes de documentos (cada uno de ellos asociado a una clase). El médico tendría que codificar los documentos que figuran más arriba en el ranking para cada clase. Esta opción se denomina *etiquetado local*, ya que se realiza localmente en cada clase. La otra opción, consiste en generar un único ranking de documentos en base a la combinación de los *m* valores de confianza asociados a un mismo documento, y se denomina *etiquetado global*. En colecciones con un número elevado de clases, como ocurre con la colección utilizada en esta investigación, el etiquetado local obliga al médico codificador a trabajar con muchos rankings diferentes. Esto supone un gran esfuerzo para la etiquetación puesto que un documento puede aparecer en múltiples rankings y el codificador debe revisarlo cada vez para asignar, o no, la etiqueta de la correspondiente clase. En cambio, la opción de etiquetado global se adapta mejor a los requisitos del trabajo a desarrollar, pues el médico codificador revisa exclusivamente un único ranking y está garantizado que un documento se lee como máximo una vez.

En este trabajo vamos a comparar algunas estrategias propuestas en [7], para la obtención de un ranking de documentos en orden decreciente en cuanto a su capacidad informativa, dentro de una estrategia de etiquetado global. Las distintas variantes se definen a través de tres dimensiones: dimensión “evidencia”, dimensión “clase” y dimensión “peso”. Cada estrategia que apliquemos va a ser una

combinación de decisiones en estas tres dimensiones, y las representaremos con una secuencia de tres letras, en donde cada letra representa una elección en cada una de las dimensiones.

Antes de entrar en detalles sobre las dimensiones, conviene aclarar terminología: dado el clasificador $\varphi^*: D \times C \rightarrow [-1,+1]$, el valor $\varphi^*(d_i,c_j)$ lo denominaremos *puntuación* de la clase c_j para el documento d_i y el valor $|\varphi^*(d_i,c_j)|$ la *confianza* de la clase c_j para el documento d_i y $sgn(\varphi^*(d_i,c_j))$ será el signo de la categoría c_j para el documento d_i.

2.1 Dimensión "evidencia"

Esta dimensión hace referencia al tipo de evidencia que utilizamos como base para la estimación de lo informativo que es un documento para una clase. Una posibilidad es utilizar el valor $|\varphi^*(d_i,c_j)|$, que representa la ***Confianza*** (**C**) del clasificador de la clase c_j con respecto al documento d_i. La intuición es que cuanto menor sea el valor de la confianza con el clasificador actual, el documento aportará más información al clasificador tras ser etiquetado (esto es, tenderemos a etiquetar los documentos sobre los que haya más duda).

La otra alternativa es usar directamente la puntuación $\varphi^*(d_i,c_j)$ como evidencia. Aquí la intuición es otra, se trata de promover documentos que son claramente ejemplos positivos de la clase porque en muchas situaciones los casos positivos son los que ayudan más en escenarios supervisados. A esta opción le denominaremos ***Puntuación*** (en inglés *Score*, ***S***).

Tenemos pues dos estrategias, *confianza* y *puntuación*, a la hora de estimar lo informativo que es un documento para una clase. La siguiente dimensión, de clase, hace referencia a cómo se combinan los *m* valores de evidencia que tienen un documento en un único valor.

2.2 Dimensión "clase"

En la dimensión "clase" lo que se pretende es generar una valoración global para cada documento independiente de la clase. Una opción consiste en maximizar la capacidad informativa esperada calculada sobre todas las clases. Esto significa que si la dimensión de evidencia es Confianza (C), para la dimensión de "clase" se tomaría $\min_{c_j \in C} |\varphi^*(d_i,c_j)|$. Si la dimensión de evidencia es Puntuación (S), para la dimensión de clase escogeríamos $\max_{c_j \in C} \varphi^*(d_i,c_j)$. Intuitivamente pretendemos que el médico codificador se concentre en documentos que se consideran de gran valor por lo menos para una clase. Esta elección se denomina ***Min / Max (M)***.

Otra opción consiste en utilizar el promedio de todos los valores obtenidos para un documento en todas las clases. De este modo seleccionaremos los documentos útiles globalmente para el conjunto de clases. Esta aproximación se llama ***Promedio*** (***Avg, A***).

Una última opción, ***Round Robin (R)***, consiste en seleccionar los documentos mejores de cada clase de la siguiente forma: se toma el mejor documento para cada clase (según dimensión de evidencia) y se crea un ranking de documentos (como mucho de tamaño *m*, pues se eliminan repetidos) ordenados decrecientemente según evidencia; a continuación, se toman los segundos documentos de cada clase, se incorporan al final del ranking (ordenados entre sí por evidencia), y así sucesivamente.

2.3 Dimensión "Peso"

La dimensión "Peso", ***Weight (W)*** tiene la función de no tratar a todas las clases por igual. Uno de los objetivos es dar más peso a las clases en donde el clasificador obtiene peores resultados. Para ello utilizamos una función de evaluación $f(\phi_j)$ que tenga un valor entre [0,1] y que nos indica qué rendimiento tiene el clasificador automático para la clase c_j. Cuando estemos trabajando con Confianza (C), multiplicaremos el valor de la confianza $|\varphi^*(d_i,c_j)|$ por la función de evaluación $f(\phi_j)$, que indica la efectividad del clasificador en la clase. Para Puntuación (S) calcularíamos el producto de $\varphi^*(d_i,c_j)$ por $(1-f(\phi_j))$. Estos ajustes sobre los valores de evidencia consiguen promover aquellas clases que no dan buen rendimiento ($f(\phi_j)$ bajo)[1]. En nuestros experimentos, al igual que se hizo en [7], utilizaremos la conocida medida F_1 para definir $f(\phi_j)$, aplicando además un suavizado de Laplace ($\varepsilon=0.05$) para evitar multiplicaciones por 0. La alternativa de no utilizar pesos para las clases la denominaremos ***No Weight (N)***.

3 Metodología para evaluar Aprendizaje Activo

Para evaluar las estrategias de AA mantendremos una colección separada de documentos de test (*TestSet*) y crearemos incrementalmente una colección de entrenamiento cuya calidad iremos contrastando mediante la construcción de un clasificador automático a partir del conjunto de entrenamiento y su evaluación contra el *TestSet*.

En el proceso de clasificación vamos a utilizar una representación vectorial de los documentos en un espacio de características. En este trabajo hemos seleccionado una representación vectorial basada en el popular esquema de pesado *tf/idf*. Para clasificar utilizaremos la metodología aplicada en [3] sobre Máquinas de Soporte Vectorial (SVM). La implementación de SVM utilizada fue SVM^{Light} [8]. Para cada clase el clasificador SVM utilizado nos devuelve un valor que representa

[1] Para confianza se multiplica por $f(\phi_j)$ porque la elección de documentos se hace en orden creciente de evidencia. Para una clase *j* con pobre rendimiento ($f(\phi_j)$ bajo), $f(\phi_j)$ multiplicado por su confianza será bajo con lo que se favorecerá la elección de los documentos. Análogamente sucede con Puntuación al multiplicarle por $(1- f(\phi_j))$ y escoger los documentos en orden decreciente.

$\varphi^*(d_i,c_j)$. La distancia al hiperplano lo interpretamos como la puntuación y su valor absoluto como la confianza.

Sea D_{Todo} el conjunto de documentos etiquetados con el que vamos a surtir a la colección de entrenamiento. El algoritmo es como sigue:

1. Seleccionamos aleatoriamente 100 documentos de D_{Todo} que denominamos D_{Train}, en donde $|D_{Train}| = 100$ y $D_{Todo} = D_{Todo} - D_{Train}$.
2. Construimos *m* clasificadores SVM, un clasificador para cada clase, utilizando D_{Train} y los evaluamos con *TestSet.*
3. Aplicamos *AA* para seleccionar 50 nuevos documentos para incorporar a D_{Train}. Para ello, según la variante utilizada:

   a. *Evidencia de Confianza* y *Clase Min (CM)*

   $\forall d_i \in D_{Todo}$ calculamos $d_{score}=\min |\varphi^*(d_i,c_j)| \; \forall c_j \in \{C_1,\ldots, C_m\}$

   Realizamos un ranking creciente por d_{score} y seleccionamos el *Top 50*.

   b. *Evidencia de Puntuación* y *Clase Max (SM)*

   $\forall d_i \in D_{Todo}$ calculamos $d_{score}=\max \varphi^*(d_i,c_j) \; \forall c_j \in \{C_1,\ldots, C_m\}$

   Realizamos un ranking decreciente por d_{score} y seleccionamos el *Top 50*.

   c. *Evidencia de Confianza* y *Clase Promedio (CA)*

   $\forall d_i \in D_{Todo}$ calculamos $d_{score}=\text{avg} |\varphi^*(d_i,c_j)| \; \forall c_j \in \{C_1,\ldots, C_m\}$

   Realizamos un ranking creciente por d_{score} y seleccionamos el *Top 50*.

   d. *Evidencia de Puntuación* y *Clase Promedio (SA)*

   $\forall d_i \in D_{Todo}$ calculamos $d_{score}=\text{avg}\, \varphi^*(d_i,c_j) \; \forall c_j \in \{C_1,\ldots, C_m\}$

   Realizamos un ranking decreciente por d_{score} y seleccionamos el *Top 50*.

   e. *Evidencia de Confianza* y *Clase Round Robin (CR)*

   $\forall c_j \in \{C_1,\ldots, C_m\}$ *calculamos* $d_{Cj}= \min |\varphi^*(d_i,c_j)| \; \forall d_i \in D_{Todo}$

   Con el documento d_{Cj} seleccionado para cada clase realizamos un ranking de documentos por orden creciente de confianza, eliminamos los duplicados y seleccionamos el *Top 50*. Si no hay 50 documentos en el ranking se tomarían los siguientes documentos seleccionados por cada clase y así sucesivamente.

   f. *Evidencia de Puntuación* y *Clase Round Robin (SR)*

$\forall c_j \in \{C_1, ..., C_m\}$ calculamos $d_{Cj} = \max \varphi^*(d_i, c_j) \ \forall d_i \in D_{Todo}$

Con el documento d_{Cj} seleccionado para cada clase realizamos un ranking de documentos por orden decreciente de confianza, eliminamos los duplicados y seleccionamos el *Top 50*. Si no hay 50 documentos en el ranking se tomarían los siguientes documentos seleccionados por cada clase y así sucesivamente.

4. Los 50 documentos obtenidos los incorporamos a D_{Train}: $D_{Train} = D_{Train} \cup$ Top50 y $D_{Todo} = D_{Todo}$ - Top50.
5. Volver al paso 2 si $|D_{Train}| < 1000$.

Si además se quiere aplicar la dimensión de Peso, lo primero es calcular F_1 para cada clase, F_1^j. Estos valores se calculan con los clasificadores construidos en el punto 2 y evaluados con *TestSet.* Y este valor lo obtenemos de la siguiente forma:

$$F_1^j = \frac{2TP_j}{2TP_j + FP_j + FN_j} \quad \forall j \in \{1, ..., m\}$$

Si $TP_j = FP_j = FN_j = 0$ entonces $F_1^j = 1$. Donde TP, FP Y FN son el número de positivos verdaderos, positivos falsos y negativos falsos, respectivamente, obtenidos por el clasificador.Al cálculo de F_1^j le aplicamos el suavizado de Laplace con un valor $\varepsilon=0.05$. Como se explicó antes estos valores se utilizan para ajustar en el proceso anterior las evidencias de los documentos en las clases.

Con este proceso incremental de creación de una colección de entrenamiento podemos elegir iterativamente los 50 mejores documentos para etiquetar. En nuestra evaluación comparamos estas estrategias de AA entre sí y, además, incluimos una alternativa aleatoria en la que se toman siempre 50 documentos al azar.

4 Experimentos

Para construir la colección realizamos un estudio de los servicios que trabajan con informes de alta mediante documento informatizado. Desde este análisis elegimos el servicio de Medicina Interna del Hospital de Conxo, que forma parte del *Complejo Hospitalario Universitario de Santiago*, España. Las razones para esta selección son el alto número de documentos, el gran tamaño de los documentos, la utilización de división de párrafos homogénea y la complejidad de los diagnósticos utilizados.
La asignación de códigos CIE-9-MC a un episodio clínico tiene los siguientes elementos importantes:

- El diagnóstico principal (DxP) es la enfermedad que tras su estudio y en el momento del alta, el médico que atendió al paciente establece como causa del ingreso.

- Los diagnósticos secundarios (DxS) se consideran aquellas enfermedades que coexisten con el DxP en el momento del ingreso o que se han desarrollado durante la estancia hospitalaria y que han influido en la duración del ingreso.

La asignación de códigos CIE-9-MC es un problema multietiqueta, pero SVM se diseñó para realizar clasificación binaria. Es posible resolver los problemas de entornos SVM multiclase, basándose en la combinación de clasificadores binarios. Estas técnicas descomponen el problema multiclase en múltiples problemas binarios. Las dos principales alternativas que se aplican en la literatura para utilizar SVM cuando el número de clases, c, es superior a dos, son *1-vs-todos* y *1-vs-1*.

- *1-vs-todos* descompone un problema multiclase con *c* clases en otros tantos problemas binarios, en donde cada una de las clases se enfrentan al resto, generando *c* clasificadores. Un nuevo documento a clasificar se ordena por aquellas clases sobre las que el clasificador maximice el margen.
- *1-vs-1* descompone el problema d*e c* clases en (c*(c-1))/2 problemas binarios, donde se crean todos los clasificadores binarios posibles de los enfrentamientos uno a uno entre las clases. Cada documento a clasificar se somete a todos estos clasificadores, y se añade un voto a la clase ganadora, resultando un ranking de clases propuesta ordenada por el número de votos.

Si añadimos el factor de trabajar con una colección con un elevado número de clases, lo más recomendable es utilizar *1-vs-todos*.

La colección está compuesta de 1823 documentos. Mediante un reparto aleatorio obtenemos 1501 documentos para la colección de entrenamiento y 322 para la colección de test. La colección de entrenamiento tiene 1238 clases diferentes y la colección de test 544 clases. En la tabla 1 detallamos las características básicas de la colección.

Tabla 1. Propiedades de la colección

	Entrenamiento	Test
# de documentos	1501	322
Tamaño	5963 Kb	1255 Kb
Media de # códigos por documento	7.06	7.05
Máximo # de códigos por documento	23	19
Media de términos por documento	519.5	508.1
Min–Max # términos en documento	64 - 1386	109 - 1419

Las métricas de evaluación requieren la existencia de un *gold standard*. En nuestro caso, conocemos los códigos correctos por cada documento de test, porque cada documento de entrenamiento o test tiene una lista de clases asignada por el

médico codificador. Por supuesto, la lista de los códigos asociados a los documentos de test sólo se usa para fines de evaluación. Adoptamos las siguientes métricas, que se han utilizado en el pasado para la evaluación de los clasificadores de documentación clínica [2,3,4]:

- *Top candidato:* proporción de casos en los que el código principal es el principal candidato (*top 1*) propuesto por el sistema automático.
- *Top 10:* proporción de casos en los que el código principal está en los primeros 10 candidatos producidos por el sistema.
- *Recall 15 y recall 20:* Nivel de recall en los primeros 15 o 20 candidatos.

Utilizamos una cadena de tres letras para cada uno de las variantes de AA. Por ejemplo, CAN es la combinación de elegir *Confianza (C)* para la dimensión "evidencia", *Avg(A)* para la dimensión "clase" y (*N)* indica que no se utiliza la dimensión "peso". Con los distintos tipos de dimensiones posibles obtenemos 12 combinaciones posibles, que generan otros tantos experimentos.

Tabla 2. Resultados *Top candidato*

#Docs.	CMN	SMN	CAN	SAN	CRN	SRN	CMW	SMW	CAW	SAW	CRW	SRW	Aleatorio
100	6.52	6.52	6.52	6.52	6.52	6.52	6.52	6.52	6.52	6.52	6.52	6.52	6.52
150	8.07	8.07	6.52	7.76	8.07	8.07	8.07	**8.39**	6.83	7.14	8.07	**8.39**	7.76
200	9.94	9.32	7.45	8.70	9.94	9.32	8.07	**10.25**	8.07	8.39	8.07	**10.25**	6.83
250	**11.49**	9.94	9.32	9.94	**11.49**	9.94	9.94	9.32	7.45	10.25	9.94	9.32	10.87
300	11.18	**11.80**	11.49	12.42	11.18	**11.80**	10.87	**11.80**	11.49	**11.80**	10.87	**11.80**	10.87
350	12.11	14.29	12.42	13.04	12.11	14.29	11.18	13.98	**14.60**	13.66	11.18	13.98	12.11
400	13.04	**15.22**	13.04	13.04	13.04	**15.22**	13.04	14.29	13.98	13.98	12.73	14.29	11.80
450	13.66	13.98	13.66	14.91	13.66	13.98	12.42	**15.53**	14.60	13.98	12.11	**15.53**	13.04
500	14.60	13.35	14.91	15.22	14.60	13.35	**15.53**	14.91	**15.53**	15.22	15.22	14.91	12.42
550	14.91	15.84	15.53	16.15	14.91	15.84	16.46	**16.77**	16.46	16.15	16.15	**16.77**	11.80
600	14.60	16.77	17.08	**17.70**	14.60	16.77	15.84	17.39	16.77	16.46	15.53	17.39	12.73
650	16.46	16.46	17.70	**20.50**	16.46	16.46	17.70	16.46	17.39	16.77	17.39	16.46	12.42
700	16.46	16.46	18.32	**18.94**	16.46	16.46	17.70	17.08	17.70	16.77	17.39	17.08	12.73
750	15.84	17.08	19.25	**19.57**	15.84	17.08	15.84	17.39	18.63	16.77	15.53	17.39	12.42
800	16.15	16.46	19.25	**20.50**	16.15	16.46	17.39	17.39	18.32	17.70	17.08	17.39	13.66
850	16.15	16.46	18.63	**19.57**	16.15	16.46	17.08	18.01	19.25	16.77	17.08	18.01	14.60
900	17.70	17.08	**19.57**	18.32	17.70	17.08	18.01	18.63	18.94	15.84	18.01	18.63	15.53
950	18.32	17.39	**19.57**	17.70	18.32	17.39	18.94	18.32	18.63	16.77	18.94	18.32	14.60
1000	18.94	17.08	**20.50**	18.32	18.94	17.08	18.63	17.39	18.94	17.70	18.63	17.39	15.22

A los 12 experimentos de AA, tenemos que añadir un nuevo experimento, por el cual obtendremos de forma aleatoria los documentos que vamos incorporando a la colección de entrenamiento. De este modo podemos comparar el rendimiento entre

los modelos de *AA* y el modelo aleatorio (que supone una técnica base realista ya que es lo que ocurre en los hospitales pues no tienen criterios específicos para codificar episodios).

Los resultados de *Top candidato* para todos los experimentos se muestran en la tabla 2. Se muestran en negrita el mejor resultado para cada tamaño del conjunto de entrenamiento.

Los experimentos nos demuestran que con AA mejoramos *Top candidato*, en relación a una selección aleatoria a lo largo de todo el proceso. Tras incorporar 1000 documentos obtenemos mediante CAN un incremento del 34,7%, en comparación con el método aleatorio.

No existen mayores diferencias entre los distintos métodos de AA pero, en todo caso, CAN, SAN y SMW resultan ligeramente más robustos. Los resultados nos manifiestan que la dimensión de "peso", con pocos documentos en la colección de entrenamiento, obtiene los mejores resultados (por ejemplo, SMW y SRW). En cambio, a medida que el número de documentos de la colección de entrenamiento se incrementa, los resultados más favorables están en los experimentos que utilizan la dimensión de clase *Avg(A)* sin necesidad de realizar pesado por clases, como lo demuestran los resultados de SAN y CAN.

Con pocos documentos en la colección de entrenamiento, SMW (Puntuación – Máximo – Peso) funciona mejor que las otras combinaciones, en cambio al aumentar el número de documentos la dimensión *Avg(A)* nos facilita los mejores valores para cualquiera de las dimensiones de "evidencia", *Puntuación (S*) o *Confianza (C)*. Podemos deducir, que con pocos documentos, aquellos que pertenecen claramente a una clase, son los más representativos para el clasificador. En cambio, a medida que el número de documentos se incrementa, la media de los valores obtenidos para todas las clases, es más informativo para el clasificador. No podemos contrastar directamente estos resultados con otros experimentos [7] de AA con colecciones multiclase, ya que las métricas utilizadas no son las mismas y las colecciones son diferentes. Para nuestra colección la dimensión de peso no funciona bien cuando se combina con la dimensión de clase *Avg(A)*, en cambio consigue mejores resultados con la combinación de las otras dimensiones. Los gráficos 1, 2, 3 y 4 nos muestra la evolución de las métricas en función del número de documentos que vamos incorporando a la colección de entrenamiento para los experimentos SAN, CAN, SMW y aleatorio

Fig. 1. Top candidato

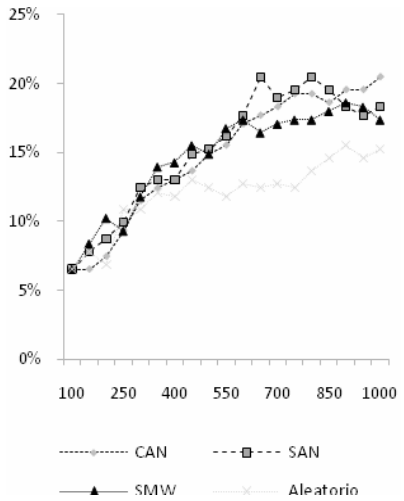


Fig. 2. Top 10

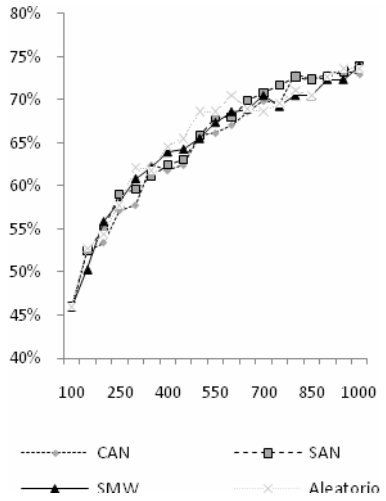


Fig. 3. Recall 15

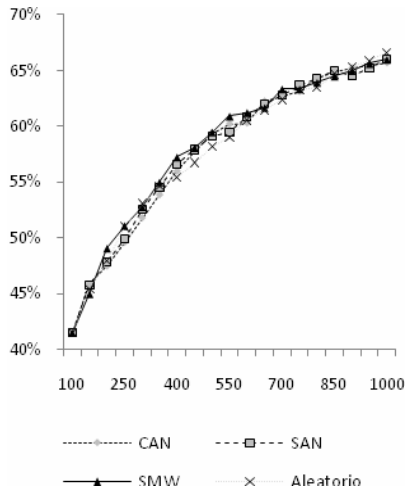
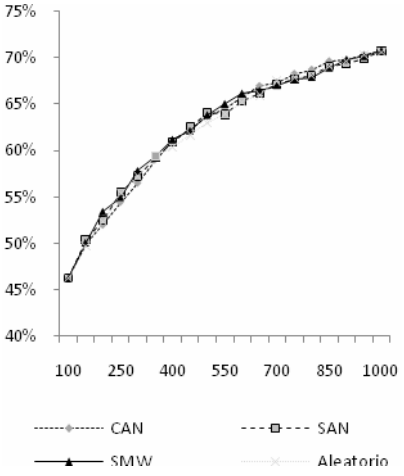


Fig. 4. Recall 20



5 Conclusiones

En los experimentos de aprendizaje activo realizados con todas las estrategias, observamos que solo *Top candidato* mejora claramente con respecto a una selección aleatoria. Para las otras métricas *Top 10, Recall 15* y *Recall 20* no se aprecia una mejora clara. Una de las posibles explicaciones es que las métricas *Top 10, Recall 15* y *Recall 20* están caracterizadas por definir un rango de posibles códigos validos sin importarle el lugar que estos ocupan dentro del ranking. Es decir para *Recall 20* tiene el mismo valor obtener un código correcto en la segunda posición dentro del rango de los 20 posibles códigos validos, que en la posición 19. Sin embargo, para el médico codificador la primera situación le demanda mucho menos esfuerzo. El aprendizaje activo mejora la posición en el rango de posibles códigos correctos, pero esta mejora no se transmite en el valor absoluto de estas métricas. Esto lo demuestra los resultados de *Top candidato* y *Top 10, y*a que, mejorando los códigos correctos que ocupan la primera posición (*Top candidato)*, este ascenso de posición del código correcto en el ranking, no se refleja en *Top 10*.

En nuestro trabajo futuro consideraremos también la posibilidad de utilizar otros clasificadores que sean más interpretables (en la línea de que el médico codificador pueda comprender mejor las sugerencias del sistema).

Agradecimientos

Agradecemos el apoyo económico de fondos FEDER, del Ministerio de Ciencia e Innovación y de la Xunta de Galicia a través de los proyectos de investigación con referencias TIN2008-06566-C04-04, 2008/068 y 07SIN005206PR.

Bibliografía

1. Clasificación internacional de enfermedades, 9ª revisión: modificación clínica CIE-9-MC. España. Ministerio de Sanidad y Consumo. ISBN 10: 84-340-1136-0
2. Larkey, L. and Croft, W. B., "Automatic Assignment of ICD9 Codes to Discharge Summaries," Center for Intelligent Information Retrieval Technical Report (1995).
3. D. Lojo, D. Losada, A. Barreiro. CIE-9-MC code Classification with knn and SVM. 3rd International Work-conference on the Interplay between Natural and Artificial Computation, IWINAC 2009, Santiago de Compostela (Spain), Jun 2009, 499-508, LNCS.
4. Larkey, L. and Croft, W. B. , "Combining Classifiers in Text Categorization," Proceedings of the 19th International Conference on Research and Development Information Retrieval (SIGIR96), Zurich, Switzerland, pp. 289-297
5. Cohn, D., Atlas, L., Ladner, R.: Improving generalization with active learning.Machine Learning 15(2), 201–221 (1994)
6. Tong, S., Koller, D.: Support vector machine active learning with applications to text classification. Journal of Machine Learning Research 2, 45–66 (2001)
7. Andrea Esuli and Fabrizio Sebastiani. Active Learning Strategies for Multi-Label Text Classification**.** Proceedings of the 31st European Conference on Information Retrieval (ECIR'09)*,* Toulouse, FR, 2009
8. Joachims, T.: Making large-scale SVM learning practical. In: Advances in Kernel Methods - Support Vector Learning. MIT press, Cambridge (1999)