5. Resultados y pruebas

El Ecode, al ser un sistema de gran tamaño y con un grado de complejidad elevado, presenta una gran cantidad de parámetros a evaluar. Debido a que el alcance de este trabajo es limitado se decidió tomar las medidas más significativas y de más fácil medición para evaluar el sistema.

Si bien éstas no son ni remotamente las únicas medidas y un trabajo de tal complejidad requeriría de una evaluación mucho más exhaustiva y considerar muchos más parámetros, se acotó la evaluación a parámetros cuyo cambio resultó de un gran impacto para el sistema: el tiempo de ejecución tanto de los etiquetadores POS como del sistema global, simplificación y reducción de líneas de código y la precisión y exhaustividad del sistema previo y el producto medido con un corpus de pruebas.

5.1 Simplificación de código

Al realizar la optimización se redujeron algunas líneas de código, lo cual impacta directamente la legibilidad y desarrollo futuro del código e, indirectamente, en el tiempo de ejecución del sistema, rendimiento analizado en otro apartado.

Parte de la simplificación del código consiste en reducir las operaciones de entrada y salida, lo cual también es medido y se presentan los resultados en cuánto se redujeron.

Para la compresión de código se midió el número de líneas de cada archivo del Ecode original y sus correspondientes en el producto final, lo que está estrechamente relacionado con el tiempo de ejecución.

En el capítulo anterior se planteó como sería compactado el código, sus optimizaciones y la reescritura de expresiones regulares. Después de haber realizado las mediciones a los archivos se obtuvieron resultados favorables, sin embargo, cabe mencionar las siguientes consideraciones.

Las diferentes acciones de optimización, que fueron planteadas por separado, muchas veces van compaginadas unas con otras, ya que en el código no es posible o no es conveniente tratarlas por separado. Por esta razón se decidió medir directamente la cantidad de líneas afectadas por cada archivo y no analizar por separado los diferentes tipos de optimizaciones.

Esta medición es relativa y lo único que indica es que se redujo la cantidad de líneas y que

el código es ahora más compacto, el impacto de estos cambios va estrechamente relacionado con el tiempo de ejecución.

5.1.1 Reducción de operaciones de entrada y salida

Un paso esencial de la simplificación del código fue la reducción al máximo de operaciones de entrada y salida; en el capítulo anterior se analizaron las lecturas y escritura al disco duro durante los módulos, en la Tabla 4.2 se muestra cuántas lecturas y escrituras de archivos se producen en el sistema.

Archivo (Módulo)	**Operaciones de Entrada (lectura de archivos)**	**Operaciones de Salida (Escritura de archivos)**
01_vds.pm	1	2
02_pvds.pm	2	2
03_td.pm	2	1
04_filtro.pm	3	2
05_arbol.pm	2	2
06_retagging.pm	1	1
07_rankingTyD.pm	2	1
08_rankingGlobal.pm	2	1
09_final.pm	7	6
01_gramaticaPOS.pm	0	0
ecode.pl	1	3

Tabla 4.2[15] Ocurrencias de operaciones de Entrada/Salida en los módulos de Ecode.

Luego de eliminar todas las entradas y salidas innecesarias al sistema, se obtuvo el siguiente resultado:

[15] Tabla del capítulo 4

Archivo (Módulo)	**Operaciones de Entrada (lectura de archivos)**	**Operaciones de Salida (Escritura de archivos)**
01_vds.pm	0	0
02_pvds.pm	0	0
03_td.pm	0	0
04_filtro.pm	0	0
05_arbol.pm	0	0
06_retagging.pm	0	0
07_rankingTyD.pm	0	0
08_rankingGlobal.pm	0	0
09_final.pm	0	0
01_gramaticaPOS.pm	0	0
ecode.pl	1	2
00_Inicia.pm	2	0
Módulo de entrada	Depende del usuario (usualmente 1)	0
Módulo de salida	0	Depende del usuario (usualmente 1)

Tabla 5.1 Ocurrencias de operaciones de Entrada/Salida en los módulos de Ecode después de su optimización

De esta manera el núcleo del sistema queda libre de lectura o escritura al disco duro, lo que genera un Ecode más rápido que sólo se ejecuta en la memoria.

5.1.2 Compresión del código

Para simplificar el código se tomaron en cuenta los cinco puntos planteados en el capítulo 4: inicialización de variables, simplificación de variables y su alcance, reescritura de expresiones regulares, compresión de estructuras de control y conjunción de ciclos similares.

El resultado final es que la compactación de líneas de código y simplificación, en general fue considerable, lo que va respaldado con los tiempos de ejecución encontrados. Para la reducción de líneas la siguiente tabla muestra el número de líneas por archivo del Ecode original:

Archivo (Módulo)	**Número de líneas Del Ecode original**	**Número de líneas en el producto final**	**% de reducción** (de acuerdo al original)
01_vds.pm	178	97	54.5
02_pvds.pm	287	195	67.94
03_td.pm	295	212	71.86
04_filtro.pm	301	241	80
05_arbol.pm	824	645	78.28
06_retagging.pm	218	240	110
07_rankingTyD.pm	394	327	83
08_rankingGlobal.pm	160	112	70
09_final.pm	125	57	45.6
01_gramaticaPOS.pm	98	92	93.88
00_Inicia.pm	Inexistente	100	130[16]
Global:	**2880**	**2318**	**80.49**

Tabla 5.2 Número de líneas de cada archivo del Ecode original y el final

[16] El 130 en realidad es ficticio y solo se agrega para obtener un valor que complete para el 80.49, en realidad este valor se refiere al porcentaje de las líneas que se sacaron de otros archivos y se agregaron 00_Inicia.pm

Al final, encontramos que la reducción global de líneas de código fue un 80.5% respecto del original.

5.2 Tiempo de ejecución

Se consideraron dos mediciones: primero, al realizar el módulo de entrada se sustituyó el programa de etiquetado POS basado en el algoritmo de Brill por el TreeTagger, el cual es mucho más rápido y es precisamente una de las mediciones a realizar: el tiempo de ejecución de los etiquetadores POS; y segundo, se midió únicamente el módulo principal de procesamiento, es decir, lo que era el Ecode original y el resultado de la optimización, para ver cuánto se logro reducir el tiempo que tarda en extraer los CDs.

Como corpus de prueba se utilizó un archivo con alrededor de 5 mil líneas que, a pesar de no ser muy grande, sí proporciona un tamaño adecuado para discernir las diferencias entre un Ecode y el otro.

5.2.1 Tiempo de ejecución de etiquetadores POS

Para la medición del tiempo de ejecución de los etiquetadores POS se utilizó un corpus (Archivo: documentos_medicion.txt) obtenido de diferentes textos elegidos al azar de la base de datos de Describe y cuenta con un total de 5297 líneas de tamaño variable, 366 *Kbytes,* que tras el filtrado del preprocesamiento sólo son validas y se procesan 4248 líneas.

Se realizó un preprocesamiento al texto antes de ambos etiquetadores, el que sería requerido para el etiquetado POS de Ecode, que principalmente consta de eliminación de líneas vacías o muy pequeñas y separación en oraciones. El tiempo de ejecución del preprocesamiento está considerado en ambas mediciones, en parte para asegurar que ambos etiquetadores recibieran la misma información y que si existe un error en el corpus de entrada, sea corregido y no intervenga en los etiquetadores.

Los resultados obtenidos se muestran en la siguiente tabla:

Corrida	Etiquetador Brill [s]	TreeTagger [s]	% tiempo que tarda el TreeTagger vs Brill
1	68.08	3.37	4.95005875
2	67.69	3.39	5.00812528
3	66.60	3.38	5.07507508
PROMEDIO	**67.46**	**3.38**	**5.0106241**

Tabla 5.3 Comparación de tiempos de ejecución de etiquetadores POS

En esta tabla, claramente se aprecia la reducción sustancial del tiempo de etiquetado POS con una reducción de 67.5 [s] a 3.38 [s], lo que se traduce en realizar el trabajo en sólo el 5% del tiempo que le tomaba originalmente.

5.2.2 Tiempo de ejecución del núcleo de Ecode

Para medir los tiempos de ejecución del procesamiento principal del Ecode y por lo tanto el impacto de la optimización que se le hizo al código, se utilizó el mismo corpus del apartado anterior (Archivo: documentos_medicion.txt), pero previamente etiquetado con POS, usando TreeTagger; sin embargo, debido a que ambos sistemas son muy rápidos y para poder tener una medida más precisa de la diferencia, se ejecutan 10 veces por cada medición.

Los resultados para el Ecode original y el producto final se muestran en la siguiente tabla:

Corrida (X 10)	Ecode original [s]	Ecode optimizado [s]	% optimización de tiempo
1	13.05	12.21	93.56
2	13.17	12.08	91.72
3	13.04	12.03	91.25
PROMEDIO	13.09	12.11	92.51

Tabla 5.4 Comparación de tiempos de ejecución del Ecode original contra el producto de la optimización.

De este modo se comprueba que se logro reducir casi en 10% el tiempo que tarda uno y otro sistema en su procesamiento principal.

5.3 Precisión y exhaustividad de los contextos extraídos contra el Ecode original

Para medir éstos parámetros se utilizó un módulo de evaluación (Archivo 10_eva.pm) del Dr. Alarcón. Asimismo, construí un corpus de 400 candidatos a contextos definitorios (Archivo: CORPUS.txt), tomando líneas al azar del corpus de trabajo del Dr. Alarcón (Alarcón 2009). Este corpus fue analizado para determinar si son o no CDs, y etiquetado por un experto, el Lic. Víctor Mijangos, que tiene conocimiento de la estructura de los contextos (Mijangos 2011) y una formación en lingüística. Con esto fue posible medir la precisión y exhaustividad de ambos sistemas.

De acuerdo con Jurafsky y Martin (2000), la precisión es una medida que se utiliza para determinar cuánta información extraída automáticamente por el sistema es correcta, mientras que la cobertura es una medida para saber cuánta de la información relevante en el texto fue extraída automáticamente.

En Alarcón (2009) se definen las fórmulas de la obtención de la precisión y exhaustividad del sistema:

$$\text{Precisión} = \frac{\text{número de respuestas válidas propuetas por el sistema}}{\text{número de respuestas propuestas por el sistema}}$$

$$\text{Cobertura} = \frac{\text{número de respuestas válidas propuetas por el sistema}}{\text{número total de respuestas en el texto}}$$

Ahora bien, pensando en el escenario de la extracción de CDs, estas fórmulas las podemos interpretar de la siguiente manera:

$$\text{Precisión} = \frac{\text{número de CDs válidos propuetos por el sistema}}{\text{número de CDs propuestos por el sistema}}$$

$$\text{Cobertura} = \frac{\text{número de CDs válidos propuetos por el sistema}}{\text{número total de CDs en el corpus}}$$

Figura 5.2 Fórmulas de precisión y exhaustividad (Alarcón 2009: 202)

Además, se adaptó el módulo de evaluación (Archivo: 10_eva.pm), el cual analiza la precisión y exhaustividad de un corpus etiquetado.

Los resultados son:

Sistema	CDs en el corpus (experto)	CDs propuestos por el sistema (automáticos)	CDs válidos propuestos por el sistema
Ecode original	150	305	126
Ecode Final	150	344	134

Tabla 5.5 Resultados globales por CDs

En números, calculando la precisión y exhaustividad:

Medición	Ecode original	Ecode Final
Precisión	0.4131147	0.3895348
Exhaustividad	0.8235394	0.8815789

Tabla 5.6 Resultados de precisión y exhaustividad

Estos resultados muestran que el Ecode original es un poco más preciso, pero el Ecode final resulta tener mayor cobertura de contextos definitorios.

5.4 Recapitulación del trabajo

De los resultados consideré varios puntos que vale la pena mencionar:

Para la simplificación del código, en cuanto a la compresión de código, se redujo en 20% el número de líneas, sin embargo, más que en la cantidad se obtuvo una optimización que permite tener código bien delimitado, modularizado y más legible para que pueda ser mejorado posteriormente. Además se redujeron prácticamente todas las escrituras Entrada y Salida a disco duro excepto en la entrada y la salida de Ecode, es decir, todo lo trabaja en memoria.

En cuanto al tiempo de ejecución, la medición se realizó con un corpus muy pequeño. Para

un corpus de un tamaño más considerable, debido a las operaciones de entrada y salida, el Ecode Final terminaría primero con una diferencia más significativa.

El cambio del etiquetador POS logró reducir 20 veces el tiempo de etiquetado, además de que TreeTagger es mucho más fácil de configurar para calibrarlo con textos afines y que sea un poco más preciso.

En el núcleo de Ecode se redujo el tiempo de ejecución en 5%, lo que si bien es poco, en corpus de gran tamaño este valor es significativo.

La precisión y exhaustividad son medidas de las cuales es importante mencionar que:

- Si bien se realizó toda una reestructura del código, la lógica del núcleo de Ecode no fue cambiada por no poseer los conocimientos lingüísticos como para tal empresa; por lo que la mayor parte de la diferencia entre los valores de los sistemas se debe a los etiquetadores POS, de nuevo este valor se puede mejorar mucho entrenando a TreeTagger para textos similares a los corpus de contextos definitorios.
- El experto tomó en cuenta todos los niveles lingüísticos para filtrar los candidatos del corpus de entrada, no solo los patrones morfo-sintácticos, sino la semántica de las palabras que no son considerados por Ecode; esto da una gran diferencia entre los contextos del experto (150) y los del Ecode original y final, 305 y 344 respectivamente. Así pues, aunque exista esta diferencia, también difieren los 2 sistemas, por lo que es posible medir qué tan precisos son los sistemas.

Además, cabe mencionar que los valores de precisión (0.53) y cobertura (0.79) plasmados en Alarcón (2009:203) difieren mucho a los obtenidos en este trabajo, tanto en el Ecode original como en el final debido a dos factores: al corpus utilizado y al experto que los clasificó. Si bien el corpus que aquí se utilizó fue obtenido de un corpus del propio Alarcón, no fue de su corpus final de pruebas, ya que no se tuvo acceso a él, pero se obtuvo de uno de sus corpus intermedios.