ERRORES ORTOGRAFICOS EN EL INGRESO EN BASES DE DATOS

Ernesto Spinak *

Resumen: Se estudian los problemas de la corrección ortográfica en el ingreso de registros en bases de datos en idioma español. Se evalúan los pros y contras de cuatro métodos de control: doble entrada, hápax legómena, trigramas y uso de diccionarios, con vistas a determinar cuáles de estos procedimientos ofrecen mejor relación de costo/resultado. El trabajo está enfocado a los procesos de ingreso por digitación, y no se analizan los errores ortográficos de los ingresos por lectura óptica.

Palabras clave: Errores ortográficos, ingresos de registros, bases de datos.

Abstract: The problems of ortographic quality control in data entry of records in databases in Spanish language are analyzed. The pros and cons of four control methods are evaluated: double entry, hapax legomena trigrammes and use of dictionaries, in view to determining which of these procedures offer a better cost/result relation. The work is focussed to the processes of manual data entry; the errors of data entry with scanners ar not analyzed.

Keywords: Ortographic errors, data entry, databases.

1 Introducción

En los últimos 20 años se ha analizado intensamente el lenguaje usando las computadoras. Estas investigaciones, objeto de especialistas de información, lingüistas, investigadores de inteligencia artificial, etc. han dado surgimiento a un área interdisciplinaria que es llamada, por algunos, lingüística computacional.

La gran mayoría de estas investigaciones se ha llevado a cabo para la lengua inglesa. Existe sin embargo muy poca o casi ninguna literatura que trate estos temas para el idioma español.

Los trabajos publicados para el español se han limitado a estudios de frecuencia de palabras y las supuestas corroboraciones o no de la ley de Zipf, sin avanzar mucho más allá de estos primeros pasos.

Otra observación que cabe hacer es que esos análisis, en general, se han llevado a cabo sobre conjuntos reducidos de datos, bases de pocos miles de registros, lo que podría poner en duda la representatividad de las conclusiones.

La importancia de este tipo de investigaciones puede comprenderse mejor si se toma en consideración que son básicas para abordar la solución, mediante procedimientos automatizados, de: correctores ortográficos; traducción automática; indización automática; análisis de estilos y paternidad literaria; compresión de datos; recuperación de información en bases de datos de texto completo, etc.

El autor de este trabajo ha estado investigando por varios años los temas antes enunciados, y se presentan en esta oportunidad algunas de las conclusiones obtenidas sobre el problema del control automatizado de errores ortográficos.

* Escuela Universitaria de Bibliotecología. Universidad de la República de Uruguay.
Recibido: 24-2-95.

2 Metodología

Se usó un computador AT-386 33 Mhz para correr los programas. El análisis se realizó sobre 12 bases de datos latinoamericanas de diferentes instituciones. Los programas utilizados se escribieron en Isispas (CDS/ISIS), dBase III + y lenguage «C». Se corrieron más de 200 horas de máquina, y fue necesario crear para los controles un diccionario de la lengua española con más de 89.000 términos, en una base en MicroIsis.

Análisis del vocabulario

Se extrajeron todas las palabras de los campos de títulos y resumen en idioma español, de 10 bases de datos bibliográficas, y además los descriptores y notas de dos tesauros. Los datos proceden de la compilación de varias instituciones latinoamericanas en el disco LILACS, producido por BIREME (1).

Se contabilizaron la cantidad total de palabras *(postings)* y la cantidad de palabras distintas (términos). Los resultados se resumen en la Tabla I.

Se generó una base llamada TOTAL, se compararon contra un diccionario preparado a tal efecto, y se clasificaron las palabras en las siguientes categorías:

E: Errores ortográficos
X: Palabras correctas en otros idiomas
S: Nombres propios, siglas, abreviaturas, tecnicismos, etc.
B: Palabras correctas

Tabla I

Nombre	*Términos*	*Postings*	*Registros*
CEPAL	8.301	190.194	8.500
CLAPAN	23.033	661.180	12.521
DOCPAL	39.410	3.005.936	32.555
ECO	3.815	21.086	+8.500
LEYES	6.682	141.309	+3.000
LILACS	75.256	2.567.761	+70.000
PPHD	13.744	131.883	+18.000
REDUC	22.307	593.552	10.246
REPIDISCA	48.118	1.893.473	35.118
SIBRA	37.676	829.587	+12.000
SINFO	13.953	413.876	5.896
TESREP	2.288	22.547	2.733
TOTAL	140.775	10.472.384	+219.000

Nota: En algunos casos no se sabe la cantidad exacta de registros por existir registros borrados dentro de la base.

Luego de eliminadas las del tipo X quedaron:

términos: 120.388
postings: 10.298.411

con la siguiente distribución:

$$E = 0{,}52\% \quad , \quad S = 1{,}92\% \quad , \quad B = 97{,}56\%$$

La frecuencia de uso de las palabras es muy disímil, tema ya sabido y que se conoce como ley de Zipf.

Se presenta el comienzo y final de la tabla de distribución al solo efecto de ilustrar ejemplos que se dan más adelante.

Palabra	*Frecuencia*	*Palabra*	*Frecuencia*
DE	968.260	LOS	212.402
LA	456.373	DEL	169.529
Y	365.724	A	150.616
EN	325.623	LAS	147.914
EL	261.033	SE	146.540

Las primeras 10 palabras sumaron 3.204.014, es decir, el 31% de todos los postings.

Las últimas palabras de la lista, las menos frecuentes son:

Frecuencia	*Palabras ≠*	*Frecuencia*	*Palabras ≠*
4	4.019	2	10.984
3	5.924	1	29.684

Esto es, 29.684 palabras distintas (25 % de los términos) se usaron sólo una vez entre todas las bases de datos.

3 Problemas de la corrección ortográfica

3.1 Análisis

El análisis de las bases de datos antes mencionadas detectó una tasa promedio de errores de digitación y ortográficos de al menos un 0.5%, en los campos de títulos y resúmenes, que escaparon inadvertidos a quienes hicieron los controles.

La tasa de errores sería un poco mayor si se hubieran examinado las palabras dentro del contexto, para identificar aquellas palabras digitadas de manera errónea pero que generaron palabras ortográficamente correctas. Además no se verificó la corrección de los términos agrupados bajo «S», esto es nombres propios, siglas, etc.

Dado que los campos analizados no son diferentes de cualesquiera otros, a saber: nombres de autores, instituciones, notas, descriptores, etc., bien pueden extrapolarse estos resultados y considerar que la tasa de errores es general a lo largo y ancho de los campos de los registros.

Los mayores impactos que causan los errores ortográficos en las bases de datos

son: pérdida de confianza en la seriedad de los datos por parte de los usuarios, decrecimiento en la tasa de recuperación, mayor dificultad en el uso del sistema.

Debe hacerse notar que la cantidad de entradas válidas, o claves, en un diccionario crece en relación logarítmica al tamaño de la base de datos. Esto es, para pequeñas bases de datos el crecimiento del diccionario es muy rápido, pero se va haciendo más lento a medida que aumentan los registros, debido a que ya han sido ingresados los términos más frecuentemente usados.

Una base de 100.000 registros contendrá aproximadamente entre el doble o el triple de términos distintos que una base de 10.000 registros, diez veces menor.

En cambio, el número de claves erróneas que se ingresan es aproximadamente proporcional al tamaño de la base de datos, y su crecimiento se mantiene a tasa constante. Con el tiempo, la proporción de entradas erróneas en el diccionario respecto a las válidas llega a ser una cifra significativa.

Durante casi 20 años la comunidad de los especialistas de información bibliográfica ha tratado de generar procedimientos para disminuir la tasa de errores cometidos que escapan inadvertidamente a los controles de ingreso, y procurar mantener la corrección formal de las bases de datos (2-6).

La literatura consultada considera cuatro métodos distintos, con variantes. Sin embargo, existen pocos análisis métricos que justifiquen la adopción de uno u otro.

A la luz del experimento realizado, se explicarán a continuación cada uno de estos cuatro métodos, con sus pros y contras. Antes será necesario analizar qué tipos de errores son los que se cometen al ingresar datos y cómo afecta esto al procedimiento de detección empleado.

3.2 Tipos de errores ortográficos posibles al digitar

En la gramática podemos considerar tres niveles de errores: el ortográfico, el sintáctico y el semántico.

Los errores ortográficos son los cometidos en palabras cuya grafía no se corresponde con un listado de autoridad o diccionario. Los errores sintácticos se producen cuando las palabras de la frase no responden a las reglas correctas de la construcción, por ejemplo la falta de concordancia del género, número o caso, o la conjugación, o la disposición incorrecta de las palabras dentro de la frase.

El error semántico se produce cuando la frase, ortográfica y sintácticamente correcta, produce un sinsentido lógico.

Un programa que determine la corrección sintáctica *(parser)*, o la corrección semántica, requiere niveles de inteligencia artificial no fácilmente disponibles al momento con carácter general, y su análisis escapa a los alcances de este trabajo.

Los errores que se analizarán son estrictamente los ortográficos.

Errores posibles producidos en la digitación

Consideremos los errores de digitación como sucesos aleatorios, donde un carácter en una secuencia de caracteres resulta no ser el correcto. A los efectos de los ejemplos supondremos que la secuencia esperada debiera ser: **ABCDE**, y cualquier otra se considerará errónea.

Sea n el largo en caracteres de una secuencia; en nuestro ejemplo $n = 5$. La cantidad de errores generados por una cierta modalidad de equivocarse será función de esa n.

Los principales tipos de errores posibles se generan por permutación, omisión, variación diagonal o repetición, de alguna letra.

Otros tipos de errores, de mucho menor frecuencia, se pueden reducir a la combinación de uno de estos tipos básicos. La inserción de un espacio en blanco o un carácter no alfabético en lugar de una letra, producirá que una palabra se quiebre en dos y generará en la mayor parte de los casos dos secuencias de letras que se considerarán ambas como errores ortográficos.

Un análisis por muestreo permitió determinar que los errores del tipo compuesto de varios simples, en conjunto, son bastante menos del 20% del total de errores, por lo que su incidencia es marginal respecto al problema general.

Esta cifra es algo menor que la mencionada por Damerau y Mays (7) en un estudio de errores mecanográficos, realizado en Inglaterra en 1989. La razón quizás sea que estos registros ya habían sido de alguna manera controlados, y los errores detectados son los que escaparon al proceso de revisión, puesto que los términos con errores compuestos se perciben más fácilmente en una corrección visual.

Analizando cada tipo de error simple se puede deducir el peso relativo que tienen en la comisión de errores. Se expresará la cantidad de errores posibles de un tipo por la letra P, D, O, V con el subíndice n que expresa el largo en letras.

Tipo de errores simples para la secuencia correcta: ABCDE.

Permutación:	BACDE	Omisión:	BCDE
	ACBDE		ACDE
($P_n = n-1$)	ABDCE	($O_n = n$)	ABCE
	ABCED		ABCD
Variación	?BCDE	Repetición:	AABCDE
diagonal:	A?CDE		ABBCDE
	AB?DE		ABCCDE
($D_n = 26 \times n$)	ABC?E	($R_n = n$)	ABCDDE
	ABCD?		ABCDEE

El signo ? puede ser cualquiera de las 27 letras del alfabeto.

En suma, una secuencia ABCDE puede producir hasta:

$$\Sigma \text{errores} = P_n + O_n + D_n + R_n = n - 1 + n + 26n + n = 29n - 1$$

secuencias no deseadas por errores de digitación.

En castellano, el largo medio de una palabra es un poco más de ocho caracteres, por lo cual esa palabra tendría 231 posibilidades de estar errada en un carácter.

Sin embargo la experiencia indica que no todos los errores son igualmente probables. Los digitadores presentan distribuciones sesgadas en los errores. Incluso un error por desconocimiento (en español confundir «c» por «s» o «z», «g» con «j») producirá errores con frecuencia mayor que la esperada aleatoriamente.

La disposición del teclado (QWERTY u otro) dará mayor probabilidad a ciertos tipos de errores por variación diagonal (D_n) que otros. Por ejemplo, es más probable digitar erróneamente en vez de una «A» cualquiera de las letras Q, Z, S que una J o una N, debido a la proximidad en el teclado. Aquellas letras que se digitan usando un mismo dedo producirán errores más frecuentes que las que se digitan con otro dedo.

En la práctica la cantidad de errores posibles esperados es mucho menor que las $29n - 1$ posibilidades, siendo aproximadamente del orden de $5{,}6n - 1$. En nuestro ejemplo significa que en la secuencia ABCDE (con $n = 5$ y teóricamente capaz de generar cualquiera de 144 errores distintos), los errores generados serán casi siempre uno de los 26 ó 27 casos más probables.

La pregunta que surge inmediatamente es:

¿Es posible diseñar un procedimiento que con niveles aceptables de certeza permita detectar cualquiera de las variaciones erróneas de una palabra que se espera correcta?

4 Métodos de control de errores de digitación

4.1 Método de doble entrada

Este sistema usado en algunas grandes instituciones, con estrategias de trabajo diseñadas hace varios años, consiste en el ingreso independiente del mismo registro por dos personas y la comparación por programa entre ambas entradas. Si existe una sola diferencia el registro es marcado para revisión.

Dado que la posibilidad de que dos personas distintas se equivoquen exactamente en el mismo carácter *(byte)* es casi nula, se puede presumir que los registros están virtualmente libres de errores de digitación.

El costo del procedimiento es mucho más alto que cualquiera de los otros considerados, porque requiere doble ingreso de datos. Además es necesario el trabajo previo de preparar la información para que pueda ser copiada por los dos digitadores independientemente.

4.2 Método de los hapax legómena

Los «hapax legómena» son los términos que aparecen una sola vez en un texto. La teoría detrás de este procedimiento se explica a continuación.

Puesto que la tasa de errores es relativamente baja (menos del 1 % del texto sin revisar), es seguro que las palabras mal escritas aparezcan sólo una vez en el texto.

Sería suficiente entonces producir un listado del texto palabra por palabra, ordenado por frecuencias, y examinar sólo aquellas palabras cuya frecuencia sea de uno o dos.

Como se ha visto más arriba, los hápax legómena dan cuenta de más de la cuarta parte del total del vocabulario, por lo cual el control manual sería muy trabajoso para textos medianamente largos.

En segundo lugar, un término que consistentemente se escribe con falta de ortografía por un digitador, no aparecerá entre los hápax legómena. Si el sistema

de ingreso permite crear campos con valores predefinidos *(default values)*, es suficiente que exista un error en uno de esos campos para que éste se arrastre a lo largo de varios registros.

4.3 Método de los trigramas

Se considera trigrama a una secuencia de tres letras cualesquiera del alfabeto. El alfabeto castellano definido en computación se compone de 27 letras: ABC DEFGHIJKLMNÑOPQRSTUVWXYZ

No se consideran letras individuales la CH, LL ni RR.

La cantidad de trigramas posibles es $27 \times 27 \times 27 = 19.683$. Sin embargo, no están todos presentes en el idioma castellano, como por ejemplo ÑÑÑ, ZZZ, SZY, etc. A su vez, de los trigramas existentes, no todos tienen la misma frecuencia. Algunos muy usados, como ADO, CON, en cambio otros muy poco usados, como KAK que sólo aparece una vez en el diccionario en la palabra kakí.

El procedimiento de verificación consiste en analizar los trigramas que componen una palabra, y si un trigrama no corresponde a uno del idioma castellano esa palabra se la considera errónea.

Debido a que la cantidad de trigramas es limitada, según se verá más adelante, todo el proceso puede correrse con los datos en la memoria RAM de la computadora, no siendo necesario el uso de extensos diccionarios, que nunca serán completos. Por llevarse todo el proceso en la memoria RAM del computador es muy rápido, de fácil programación y de bajo costo.

Para determinar las necesidades reales y limitaciones, se analizaron los trigramas existentes y sus frecuencias en un diccionario de 88.751 palabras, compilado a partir de las bases de datos. Este incluye algunos nombres de países y abreviaturas comunes (etc., ADN).

Estas palabras generaron 662.413 trigramas, siendo de éstos 3.636 distintos. Vale decir, de los 19.683 trigramas posibles, sólo poco más de 3.600 son usados en el idioma castellano.

Si se eliminan los nombres de países, abreviaturas y siglas del diccionario, desaparecen una buena cantidad de estos trigramas cuya frecuencia es sólo 1, y la cantidad desciende a 3.378 trigramas distintos.

La frecuencia de uso es muy disímil, siendo los más frecuentes: ADO 7.765 veces, ENT 5.856 veces, ADA 5.678 veces, etc. En cambio otros aparecen dos veces en todo el diccionario (DOÑ, AZT...).

Si se realiza el análisis, ponderando con la frecuencia del uso de las palabras en el idioma escrito corriente, muchos de los posibles trigramas no es necesario que sean considerados, por aparecer únicamente en palabras muy raramente usadas.

Entonces es posible cubrir el 99,99 % del vocabulario con 3.378 trigramas; el 99 % con 2.369 trigramas; 98 % con 2.061, etc.

Se considera que 3.378 trigramas permiten una cobertura casi perfecta del idioma, y se tomaron éstos para los experimentos.

El largo medio de las palabras en español es de 8-9 letras; esto genera 6-7 trigramas. Usando búsqueda por bipartición dentro de un vector de trigramas, una palabra para ser aceptada requiere hacer 60 comparaciones, y una cantidad menor

para ser rechazada. Este procedimiento es muy lento, por lo cual se desarrolló otro algoritmo más efectivo que se explica a continuación.

El método consiste en almacenar una matriz de 27 × 27, (esto es un vector de 729 posiciones en memoria), donde la clave de acceso es el valor ASCII de las dos primeras letras del trigrama. La primera letra es el número de fila, la segunda letra el número de la columna, y el contenido de cada elemento de la matriz son las letras que forman un trigrama con ese digrama. Esta matriz ocupa menos de 4 Kb de memoria. Ejemplo:

(1,1)	AA:	No ocupada porque no exite trigrama que empiece con AA
(1,2)	AB:	ACDEIJLNORSUY
(1,3)	AC:	ACEHILMNORSTU
(1,4)	AD:	AEHIJLMOQRSUVY
(1,5)	AE:	CDLNQRST
(26,20)	ZT:	E
(26,21)	ZU:	ACDEFLMNRTZÑ

Esto significa que el digrama AE forma trigrama con las letras CDLNQRST, en cambio el digrama ZT sólo forma trigrama con la letra E (en aZTeca) en idioma español.

El programa de verificación, en seudocódigo, es el siguiente:

1 Se toma palabra siguiente en el texto
1.1 Si es final de texto → FIN del proceso.
2 Para $i = 1$ hasta hasta $i =$ (largo de palabra -2).
2.1 Si i > largo de palabra − 2 se va a 1.
2.2 fila_matriz ← Valor ASCII de letra iésima -64.
2.3 columna_matriz ← Valor ASCII letra (iésima $+1$) -64.
2.4 Se busca letra (iésima + 2) en Matriz (fila, columna).
2.4.1 Si la letra está se vuelve a 2.
2.4.2 No está, se marca la palabra como errónea y se vuelve a 1.

El algoritmo requiere hacer sólo una comparación por trigrama a una tabla de acceso directo, eliminando la búsqueda por bipartición.

Este procedimiento es mucho más rápido que comparar contra palabras completas en un diccionario extenso. La razón es que la búsqueda mediante diccionario deberá hacerse a través de índices que no caben completamente en memoria lo que demandará al programa realizar varios accesos a disco antes de localizar el término.

En un diccionario de 100.000 términos, usando B*tree es necesario una media de 5 accesos a disco antes de ubicar el término en una hoja del árbol.

Puede optimizarse el procedimiento del diccionario en disco, manteniendo un *Stop-list* (lista de palabras más frecuentes) en memoria de manera de validar las palabras más frecuentes sin recurrir al uso del diccionario.

Con un *Stop-list* de 512 palabras puede resolverse el 67 % de las consultas, con uno de 64 palabras el 48 % de las consultas.

Usando bipartición, son necesarias 6 consultas en una lista de 64 términos, lo que para el 48 % de las palabras el procedimiento es un poco más rápido que el uso de trigramas.

Sin embargo, el uso del *Stop-list* hace más lenta la validación del 52 % restante

debido a que son necesarios los 4 a 6 accesos a disco, más las 6 comparaciones infructuosas en memoria antes de verificar la palabra, arrojando un promedio de tiempo total mayor que el procedimiento de trigramas

Nada se gana aumentando el *Stop-list* a 256 palabras, que cubren el 60 % con 8 comparaciones, porque hace aún más lenta la validación del 40 % restante por la razón antes mencionada.

La dificultad principal del método de los trigramas estriba en que existe una gran cantidad de combinaciones de trigramas válidos que no generan palabras existentes. Dicho al revés, es posible equivocarse cambiando letras y generar combinaciones de trigramas que serán consideradas palabras válidas por este procedimiento.

Se analizó sobre una muestra de 2340 palabras cuál es la posibilidad de generar palabras no existentes pero de trigramas válidos, a partir de palabras existentes cometiendo los errores de tipo P, V, D, O, antes definidos.

El programa generó a cada palabra de la muestra todas las variaciones posibles de las antes descritas y se consultaron en un caso contra los trigramas, y en el segundo caso contra el diccionario. Los resultados de los trigramas se ven en la Tabla II, de la que se desprenden las siguientes conclusiones:

1. Usando el procedimiento de los trigramas un 38 % de los errores serán aceptados como palabras correctas.
2. La tasa de aceptación es independiente del largo de las palabras.
3. La tasa de errores ortográficos detectada en las bases de datos fue del 5,3 por mil.
4. Componiendo esta tasa con el porcentaje de errores no detectados, es de esperar que el 2 por mil pase inadvertido.

Tabla II

Consulta con trigramas

Palabras distintas: 2.340 - Consultas generadas: 120.047

Largo	*Cant.*	*Perm.*	*Omis.*	*Var.*	*Rep.*	*Total*	*Tot/cons.*	*%*
2	4	0	0	0	1	1	8	12
3	6	8	0	19	5	32	76	42
4	34	38	103	147	22	310	734	42
5	104	170	374	551	80	1.175	2.847	41
6	169	320	689	1.059	124	2.192	5.566	39
7	259	575	1.234	1.864	241	3.914	10.043	39
8	360	864	1.907	3.035	352	6.158	15.984	39
9	390	1.088	2.302	3.710	444	7.544	19.577	39
10	339	975	2.181	3.488	419	7.063	18.912	37
11	269	840	1.882	3.078	349	6.149	16.507	37
12	170	587	1.256	2.088	201	4.132	11.390	36
13	111	431	917	1.439	153	2.940	8.039	37
14	60	266	523	804	95	1.688	4.666	36
15	38	188	351	548	69	1.156	3.192	36
16	17	79	166	281	30	556	1.515	37
17	6	23	69	110	11	213	571	37
18	2	12	25	39	4	80	202	40
19	2	15	28	34	6	83	218	38
Total	2.340	6.479	14.007	22.294	2.606	45.386	120.047	38

4.4 Método del diccionario

El procedimiento consiste en comparar cada palabra del texto contra las existentes en un diccionario. Si está presente se acepta, si no está se la marca como probable errónea.

El principal problema está en determinar cuál es el tamaño óptimo que debe tener el diccionario.

En el idioma castellano, a diferencia de otros como el inglés, para cada palabra existe una gran cantidad de derivadas por género, número, o flexión verbal. El vocabulario puede ser tan grande como se quiera.

Considere a vía de ejemplo la palabra «considerar»:

Sust.: CONSIDERACION, consideraciones
Adv. : CONSIDERABLEMENTE
Verbo: CONSIDERAR, considerarme, considerando, considera, consideran, consideremos, consideró, consideraron, consideraba, consideraban, consideramos, considerándose, considerarla, considerarlas, considerarlo, considerarlos, considere, consideren, considerarse, considerándola, considerándolas, considerándole, considerará, considerarán, consideraría, considerarían, considerarle, considerarles, considerársela, considerárselas, considerárselo, considerárselos, considerársele, considerárseles, considerado, considerada, considerados, consideradas...

Naturalmente que esto no agota la lista de todas las variantes de esa palabra; faltan tiempos verbales y personas. Es suficiente que aparezca en el texto alguna variación no contemplada para que se la considere como errónea.

Un procedimiento lexicográfico que analice radicales más los prefijos y sufijos introduce complicaciones de software, mayores tiempos de ejecución, y adolece de serios errores, como se pueden constatar observando el desempeño de programas de traducción automática.

A medida que crece el diccionario disminuye la posibilidad de rechazar una palabra correcta como errónea, pero a su vez agrega otras dificultades.

En primer lugar exige mayor espacio de almacenamiento en disco, lo que puede llegar a decenas de megabytes.

En segundo lugar, a medida que crece el diccionario se hace más lenta la búsqueda. En un diccionario almacenado con estructura de B * tree +, (el almacenamiento más efectivo para este tipo de recuperación) de 80 a 90 mil términos, es necesario hacer entre 4 y 6 accesos a disco antes de determinar si la palabra existe o no. Estos tiempos se pueden mejorar muy poco manteniendo en memoria un Stoplist de las palabras más frecuentemente usadas.

En tercer lugar, a medida que crece el diccionario, cada vez con palabras de menor frecuencia, aumenta la probabilidad de aceptar una palabra mal escrita (por permutación, variación diagonal, omisión, etc.) como correcta.

Al igual que para los trigramas, se analizó sobre una muestra de 2340 palabras cuál es la posibilidad de generar errores de digitación de los tipos P, V, D, O, antes definidos, pero sin embargo seguir siendo una palabra válida. El resultado se presenta en la Tabla III de la que se desprenden las siguientes conclusiones:

1. Usando el procedimiento del diccionario un 1,5 % de los errores cometidos serán aceptados como palabras correctas.

2. La tasa de aceptación de palabras erróneas decrece con el largo de éstas.
3. La tasa de errores ortográficos detectada fue del 5,3 por mil.
4. Componiendo esta tasa con el porcentaje de errores no detectados, es de esperar que el 0,1 por mil pase inadvertido.

Tabla III

Consulta con diccionario

Palabras distintas: 2.340 - Consultas generadas: 120.101

Largo	*Cant.*	*Perm.*	*Omis.*	*Var.*	*Rep.*	*Total*	*Tot/cons.*	*%*
2	4	2	8	6	2	12	44	27,3
3	6	3	11	3	0	17	94	18,0
4	34	4	14	24	2	42	734	5,7
5	104	12	101	41	0	154	2.847	5,4
6	169	12	126	31	1	169	5.566	3,0
7	259	21	137	20	2	178	10.043	1,8
8	360	31	212	18	1	261	15.984	1,6
9	390	26	226	9	0	261	19.577	1,3
10	339	21	211	1	0	233	18.912	1,2
11	269	20	137	2	0	159	16.507	1,0
12	170	15	98	1	0	114	11.390	1,0
13	111	10	45	0	0	55	8.039	0,7
14	60	3	29	0	0	32	4.666	0,7
15	38	3	10	0	0	13	3.192	0,4
16	17	1	8	0	0	9	1.515	0,6
17	6	0	3	0	0	3	571	0,5
18	2	0	2	0	0	2	202	1,0
19	2	1	0	0	0	1	218	0,5
Total	2.340	184	1.379	157	8	1.728	120.101	1,5

El método es adecuado para verificar términos cuyo largo permite suficiente redundancia, de manera que los elementos válidos del dominio de validación no sean lo suficientemente próximos.

Para aclarar esta última idea pongamos por caso la validación de los códigos ISO para nombres de países. El código de dos letras genera un dominio de posibilidades de $27 \times 27 = 729$, de los cuales son válidos 221, es decir, el 30 %. El código de tres letras genera un dominio de posibilidades de 19.683, del cual sólo el 1 % es válido. Esto es, son menos próximos.

Se procedió a hacer un experimento de generación de errores a partir de los códigos, cotejando en un caso contra un diccionario de códigos de dos letras, y en otro de tres letras.

Códigos válidos de 2 letras: 221 - Consultas generadas: 1.799

Códigos válidos de 3 letras: 189 - Consultas generadas: 1.833

Largo	*Cant.*	*Perm.*	*Diag.*	*Total*	*Tot/cons*	*%*
2	221	73	446	519	1.799	28,85
3	189	11	41	52	1.833	2,84

El uso del método del diccionario no es suficiente para asegurar la exactitud de los términos, cuando éstos no tienen la suficiente redundancia. En el caso particular de los códigos de países, el código de dos letras ofrece poca seguridad de ser detectados los errores con un diccionario.

4.5 Ingresos por lectura óptica

Cuando el ingreso de datos no es mediante la digitación, sino por dispositivos de lectura óptica *(scanners)* y el reconocimiento de caracteres *(Optical Character Recognition - OCR)*, los errores se generan por otros factores, y esto modifica un tanto los procedimientos de corrección.

En primer lugar el método de doble entrada no es aplicable, porque los errores no se deben al fallo humano, sino a las características tipográficas de la página leída y la insuficiencia del software de reconocimiento. Al ser un proceso mecanicista, los errores se repetirán en los mismos lugares.

Esto deja sólo a los métodos de trigramas y diccionario como aplicables. La elección de uno u otro dependerá de si la base de datos está definida con estructura de campos indizada, o sólo de texto completo, con índices sobre campos especiales por separado.

5 Comparación de los métodos

— Doble entrada

PROS: Produce textos libres de errores ortográficos; el procedimiento es sencillo así como el programa de comparación.

CONS: El costo es alto; no es aplicable para ingresos por lectura óptica.

— Hapax legómena

PROS: El procedimiento es el más sencillo.

CONS: No asegura que los errores sin detectar estén por debajo de una tasa aceptable. Al ser un procedimiento manual es trabajoso y lento. No asegura que no se escapen errores en la revisión manual.

— Diccionario

PROS: Este procedimiento aceptará muy pocas palabras erróneamente digitadas. Es relativamente rápido. En el caso de la validación de nombres de autores o descriptores, es el mejor procedimiento posible.

CONS: Siempre existirán palabras que no serán reconocidas, cualquiera que sea el tamaño del diccionario. Se requiere mucho espacio de almacenamiento en disco. No es bueno para códigos de baja redundancia.

— Trigramas

PROS: Todo el proceso se corre en la memoria RAM no siendo necesario accesos a disco. Tampoco es necesario contar con grandes diccionarios que requieren varios megabytes de almacenamiento. No rechaza ninguna palabra correctamente escrita. Es el procedimiento más

económico y rápido de los analizados. Aplicable a lectura óptica de textos completos, cuyos campos no son los que generan índices.

CONS: El umbral de palabras aceptadas como correctas, pero en realidad siendo erróneas, es alto comparado con el método del diccionario.

Bibliografía

1. LILACS, etc.
2. ZAMORA, A. Automatic detection and correction of spelling errors in a large data base. *Journal of the American Society for Information Science*, 1980, vol. 30, núm. 1, p. 51-57.
3. HUDSON, J. Bibliographic record maintenance in the online environment. *Information Technology and Libraries*, 1984, vol. 3, núm. 4, p. 388-393.
4. TROY, W. Better than Webster: Thumbing through WP'spell utilities. *Wordperfect: the magazine*, 1991, Jan, p. 83-6.
5. POLLOCK, J. J.; ZAMORA, A. Collection and characterization of spelling errors in scientific and scholarly text. *Journal of the American Society for Information Science*, 1983, vol. 34, núm. 1, p. 51-58.
6. O'NEILL, E. T.; VIZINE-GOETZ, D. The impact of spelling errors on databases and indexes. *Proceedings, National Online Meeting*, 1987, May 9-11, p. 313-320, New York.
7. DAMERAU, J. F.; MAYS, E. An examination of undetected typing errors. *Information Processing & Management*, 1989, vol. 25, núm. 6, p. 659-664.