Consultas con Errores Ortográficos en RI Multilingüe: Análisis y Tratamiento*

Misspelled Queries in Cross-Language IR: Analysis and Management

David Vilares Calvo
Depto. de Computación
Universidade da Coruña
Campus de Elviña s/n
15071 – A Coruña
david.vilares@udc.es

Adrián Blanco González
Depto. de Informática
Universidade de Vigo
Campus As Lagoas s/n
32004 – Ourense
adbgonzalez@uvigo.es

Jesús Vilares Ferro
Depto. de Computación
Universidade da Coruña
Campus de Elviña s/n
15071 – A Coruña
jvilares@udc.es

Resumen: Este artículo estudia el impacto de los errores ortográficos en las consultas sobre el rendimiento de los sistemas de recuperación de información multilingüe, proponiendo dos estrategias para su tratamiento: el empleo de técnicas de corrección ortográfica automática y la utilización de n-gramas de caracteres como términos índice y unidad de traducción, para así aprovecharnos de su robustez inherente. Los resultados demuestran la sensibilidad de estos sistemas frente a dichos errores así como la efectividad de las soluciones propuestas. Hasta donde alcanza nuestro conocimiento no existen trabajos similares en el ámbito multilingüe.
Palabras clave: Recuperación de información multilingüe; traducción automática; errores ortográficos; corrección ortográfica; n-gramas de caracteres.

Abstract: This paper studies the impact of misspelled queries on the performance of Cross-Language Information Retrieval systems and proposes two strategies for dealing with them: the use of automatic spelling correction techniques and the use of character n-grams both as index terms and translation units, thus allowing to take advantage of their inherent robustness. Our results demonstrate the sensitivity of these systems to such errors and the effectiveness of the proposed solutions. To the best of our knowledge there are no similar jobs in the cross-language field.
Keywords: Cross-language information retrieval; machine translation; misspellings; spelling correction; character n-grams.

1. Introducción

En el marco actual de globalización de la red muchas veces un documento relevante para el usuario está escrito en una lengua diferente a la suya. Como respuesta a esta problemática surge la *recuperación de información multilingüe* (RIM)[1] (Nie, 2010), un caso particular dentro de la *recuperación de información* (RI) en el que consultas y documentos están en idiomas diferentes.

Para ello la mayoría de dichos sistemas introducen algún tipo de *fase de traducción* intermedia que permita reducir el problema original a la clásica RI monolingüe con consultas y documentos en el mismo idioma. Debido a limitaciones prácticas suele optarse por traducir las consultas de su idioma original (denominado *origen*) al de los documentos (denominado *destino*) (Nie, 2010).

Fruto de esta misma globalización cada vez es más necesario disponer de sistemas capaces de operar sobre textos con *errores ortográficos*,[2] en particular en el caso de las consultas (Guo et al., 2008). Esto se debe a que los modelos formales de RI fueron diseñados para operar sobre textos sin errores, por lo que su presencia dañará substancialmente el rendimiento. Hablaremos entonces de *recuperación de información tolerante a errores*[3] (Manning, Raghavan, y Schütze, 2008, Cap. 3).

En este contexto nuestro trabajo aborda el

* Trabajo parcialmente subvencionado por el Ministerio de Economía y Competitividad y FEDER (proyectos TIN2010-18552-C03-01 y TIN2010-18552-C03-02) y por la Xunta de Galicia (ayudas CN 2012/008, CN 2012/317 y CN 2012/319).

[1] *Cross-language information retrieval* (CLIR)

[2] Tanto aquéllos fruto del desconocimiento de la ortografía como errores tipográficos o producto del ruido durante su generación (ej. OCR) (Kukich, 1992).

[3] *Tolerant Information Retrieval*.

estudio del impacto de los errores ortográficos en las consultas sobre el proceso de recuperación multilingüe así como el diseño de entornos robustos capaces de operar en ese contexto.

El tratamiento de consultas mal escritas suele basarse en sustituir o modificar el algoritmo de búsqueda de correspondencias exactas original para permitir correspondencias aproximadas. Conforme al estado del arte, consideramos dos estrategias genéricas diferentes (Manning, Raghavan, y Schütze, 2008): una que opera a nivel de palabra y otra a nivel de subpalabra.

La primera de éstas opera a nivel de palabra y consiste en añadir una fase de preprocesamiento para la corrección de los errores ortográficos de la consulta empleando técnicas de *procesamiento del lenguaje natural* (PLN) basadas en diccionarios. Debemos señalar que a diferencia de otros ámbitos clásicos de aplicación de los sistemas de corrección (ej. procesadores de texto), en el caso de RI se requieren soluciones que permitan un tratamiento totalmente automático del error (Kukich, 1992) sin necesidad de la intervención del usuario. Se pueden distinguir dos enfoques: la *corrección de palabras aisladas* en la cual se intenta corregir cada palabra por separado (Savary, 2002), y el aprovechamiento de la información lingüística de su contexto para la corrección (Otero, Graña, y Vilares, 2007).

Una segunda estrategia consiste en emplear n-gramas de caracteres como unidad de procesamiento en lugar de palabras (McNamee y Mayfield, 2004a; Robertson y Willett, 1998).

En este trabajo probaremos ambas aproximaciones con errores humanos reales en un contexto de recuperación *de-español-a-inglés* (consultas en español y documentos en inglés). Hasta donde alcanza nuestro conocimiento no existen trabajos similares con este grado de detalle en el ámbito multilingüe.

La estructura del artículo es como sigue. La Sección 2 aborda nuestras propuestas basadas en corrección, mientras que la Sección 3 describe nuestra propuesta basada en el empleo de n-gramas de caracteres. La Sección 4 detalla nuestro metodología de prueba, obteniendo los resultados recogidos en la Sección 5. Finalmente, la Sección 6 presenta nuestras conclusiones y propuestas de trabajo futuro.

2. *Aproximaciones basadas en corrección ortográfica*

La primera de las estrategias contempladas pasa por preprocesar la consulta empleando técnicas de corrección automática basadas en PLN para detectar y corregir sus errores ortográficos, la cual ha sido ya aplicada con éxito en RI monolingüe (Vilares, Vilares, y Otero, 2011). En nuestro contexto actual la consulta inicial es preprocesada y, una vez corregida, es traducida aplicando técnicas de *traducción automática* (TA)[4] y luego lanzada contra el motor de recuperación. Asimismo se han considerado dos posibles técnicas corrección, aislada y contextual, que describimos a continuación.

2.1. Corrección aislada

Como punto de partida aplicaremos el algoritmo de reparación global propuesto por Savary (2002), capaz de encontrar todas las palabras cuya distancia de edición (Levenshtein, 1966) con la errónea sea mínima; esto es, el número de *operaciones de edición*[5] a aplicar para transformar una cadena en otra.

Este algoritmo tiene como núcleo un *autómata finito* (AF) que reconoce el léxico del idioma considerado. Para cada palabra a procesar el AF intenta reconocerla intentando ir desde el estado inicial a uno final a través de las transiciones etiquetadas con los caracteres de la cadena de entrada. Si el AF se detiene en un estado por no haber transiciones de salida etiquetadas con el siguiente carácter de la entrada, es que se ha detectado un error ortográfico. Pasamos entonces a aplicar sobre la configuración actual del autómata cuatro posibles *hipótesis de reparación* elemental, cada una de ellas correspondientes a una operación elemental (inserción, borrado, sustitución y transposición) y con un coste asociado para así intentar alcanzar una nueva configuración que nos permita continuar con el proceso de reconocimiento. Dichas operaciones se aplican recursivamente hasta alcanzar una configuración correcta. El algoritmo también reduce dinámicamente el espacio de búsqueda, quedándose en todo momento únicamente con las correcciones mínimas y tratando de alcanzar la primera solución tan pronto como sea posible.

[4] *Machine translation* (MT).

[5] Inserción, borrado o substitución de un carácter, o transposición de dos caracteres contiguos.

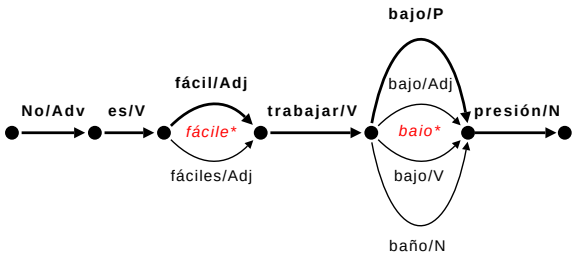

Figura 1: Alternativas de corrección ortográfica representadas en una retícula (secuencia correcta resaltada).

Desafortunadamente, este algoritmo puede devolver varias correcciones candidatas si existen varias palabras a distancia mínima de edición de la palabra original. Tomemos como ejemplo la frase *"No es fácile* trabajar baio* presión"*,[6] representada junto con sus posibles correcciones en la Figura 1; en este caso el algoritmo devolvería dos posibles correcciones candidatas para *"fácile*"* (*"fácil"* y *"fáciles"*) y otros dos para *"baio*"* (*"bajo"* y *"baño"*). Asimismo este algoritmo está limitado a aquellos errores correspondientes a palabras no válidas, por lo que puede fallar a la hora de detectar errores que dan lugar a otras palabras válidas del idioma, como podría ser el caso de la consulta *"materiales compuestos ligueros*"*[7] pues tomados por separado cada uno de sus términos es válido.

2.2. Corrección contextual

Existe a su vez una extensión al algoritmo anterior, que denominaremos de *corrección contextual*, que permite emplear el contexto lingüístico de la palabra a corregir para resolver las limitaciones del algoritmo original (Otero, Graña, y Vilares, 2007). Para ello se utiliza la información lingüística contextual embebida en un proceso de etiquetación con el fin de podar las correcciones candidatas de tal forma que sólo se acepten aquéllas que encajen en el contexto morfosintáctico de la palabra a corregir.

Este modelo emplea un etiquetador morfosintáctico estocástico basado en una extensión dinámica del algoritmo de Viterbi sobre *Modelos Ocultos de Markov* (Graña, Alonso, y Vilares, 2002) de segundo orden que se aplica sobre retículas en lugar de enrejados, haciéndola mucho más flexible al permitirnos representar un par *palabra/etiqueta* en cada arco, y luego calcular la secuencia más probable mediante una adaptación del algoritmo de Viterbi, como se muestra en la Figura 1.[8]

Así, y ya restringiéndonos al ejemplo de dicha figura, las correcciones devueltas serían únicamente las correspondientes a la secuencia de etiquetas correcta: *"fácil"* (`Adj`) para *"fácile*"* y *"bajo"* (`P`) para *"baio*"*.

3. Aproximaciones basadas en n-gramas

Un *n-grama de caracteres* es una secuencia de n caracteres dentro de una palabra. De esta forma `tomato` se descompone en los 3-gramas superpuestos: `-tom-`, `-oma-`, `-mat-` y `-ato-`. Las ventajas que aporta el tratamiento a nivel de n-grama —simplicidad, eficiencia, robustez, completitud e independencia del dominio— lo han convertido en una técnica habitual en el procesamiento de textos (Robertson y Willett, 1998; Vilares, Vilares, y Otero, 2011). En el caso concreto de RI, los sistemas clásicos suelen emplear conocimiento y recursos lingüísticos tales como listas de *stopwords*, *stemmers*, lexicones, tesauros, etiquetadores, etc. (McNamee y Mayfield, 2004a), mientras que la *tokenización* en n-gramas no emplea ninguno: el texto es meramente dividido en n-gramas superpuestos (McNamee y Mayfield, 2004b), que son procesados por el motor de recuperación como cualquier otro término. Se trata, pues, de una aproximación independiente del idioma y del dominio, donde además el empleo de correspondencias a nivel de n-grama constituye en sí mismo un mecanismo de normalización que permite trabajar con gran variedad de idiomas sin procesamiento alguno a mayores (McNamee y Mayfield, 2004b; Robertson y Willett, 1998; McNamee y Mayfield, 2004a). Es además un proceso robusto, debido a la redundancia introducida por el proceso de *tokenización* (Vilares, Vilares, y Otero, 2011).

3.1. RIM basada en n-gramas

En el caso de RIM, sin embargo, tales ventajas quedan comprometidas por el proceso de traducción, que debe hacerse a nivel de palabra o frase, pudiendo *tokenizarse* la consulta en n-gramas sólo tras traducirse siendo

[6] Por *"No es fácil trabajar bajo presión"* y donde los asteriscos señalan palabras mal escritas.

[7] Por *"materiales compuestos ligeros"*.

[8] Si bien sería posible emplear el algoritmo original basado en enrejados, su aplicación sería mucho más compleja y costosa (Graña, Barcala, y Vilares, 2002).

además el proceso de traducción muy sensible a los errores ortográficos, palabras desconocidas, falta de recursos lingüísticos apropiados, etc. De este modo, por ejemplo, una palabra mal escrita como "fácile*" no podría ser traducida correctamente, obteniendo[9] "facile*" en lugar de "easy", perdiendo así el procesamiento posterior con n-gramas su capacidad de realizar correspondencias aproximadas. Sólo si también se pudiese traducir a nivel de n-grama[10] podrían beneficiarse plenamente los sistemas de RIM de las ventajas derivadas del uso de n-gramas.

McNamee y Mayfield (2004b) fueron pioneros en este campo, empleando para ello un algoritmo de traducción de n-gramas basado en el alineamiento de corpus paralelos a nivel de n-grama mediante técnicas estadísticas. Posteriormente, Vilares, Oakes, y Vilares (2007) desarrollaron un sistema alternativo que difiere en el proceso de generación de alineamientos, preservando las bondades del sistema previo pero solventando sus principales desventajas.

Estas aproximaciones permiten extender las ventajas del empleo de n-gramas como unidad de procesamiento al proceso de traducción y, consecuentemente, también a los sistemas de RIM, pudiendo así evitar algunas de las limitaciones de las técnicas clásicas, tales como la necesidad de normalizar las palabras o la imposibilidad de traducir palabras desconocidas. Además, al no emplear ningún tipo de procesamiento particular dependiente del idioma, puede aplicarse cuando la disponibilidad de recursos lingüísticos es reducida, lo que, en contra de lo que pueda parecer, no es infrecuente incluso para los principales idiomas europeos (Rehm y Uszkoreit, 2011).

Seguidamente describiremos brevemente el sistema de Vilares, Oakes, y Vilares (2007), que será el que empleemos.

3.2. Traducción de n-gramas

El algoritmo de alineamiento de n-gramas, sobre el que se asienta el sistema de traducción, consta de dos fases. Primero se alinea a nivel de palabra un corpus paralelo de los idiomas origen y destino deseados empleando la herramienta estadística GIZA++ (Och y Ney, 2003), obteniendo las probabilidades de traducción entre las palabras de ambos idiomas. Dicho alineamiento será bidireccional (Koehn, Och, y Marcu, 2003), aceptando un alineamiento $idiomaOrigen \rightarrow idiomaDestino$ (w_s, w_t), donde w_s denota la palabra en lengua origen y w_t su traducción candidata, sólo si existe también el alineamiento inverso $idiomaDestino \rightarrow idiomaOrigen$ (w_t, w_s). Se desecharán también aquellos alineamientos con probabilidad menor de 0,15.

A continuación, en la segunda fase del algoritmo, se realiza el alineamiento a nivel de n-gramas propiamente dicho calculando medidas estadísticas de asociación (Dale, Moisi, y Somers, 2000, Cap. 24) entre los n-gramas contenidos en los pares de palabras alineadas en la fase anterior. Sin embargo, a la hora de realizar los cálculos deberá ponderarse la frecuencia de las observaciones de las coocurrencias de n-gramas en base a las probabilidades de los alineamientos de las palabras que los contienen. Esto se debe a que no se trata de coocurrencias *reales*, sino únicamente *probables*, de forma que una misma palabra origen puede estar alineada con más de una traducción candidata. Tomemos como ejemplo el caso de las palabras en español `leche` y `lechoso`, y las inglesas `milk`, `milky` y `tomato`; un posible alineamiento a nivel de palabra sería:

w_s	w_t	**prob.**
`leche`	`milk`	0,98
`lechoso`	`milky`	0,92
`leche`	`tomato`	0,15

En este caso la frecuencia de coocurrencia del par de n-gramas *(`-lech-`, `-milk-`)* no sería 2, sino 1,90, ya que si bien dicho par coocurre en dos alineamientos a nivel de palabra, *(`leche`, `milk`)* y *(`lechoso`, `milky`)*, dichas coocurrencias se ponderan en base a las probabilidades de sus alineamientos:

$$0{,}98\ (\texttt{leche, milk}) + 0{,}92\ (\texttt{lechoso, milky}) = \mathbf{1{,}90}$$

4. Metodología

4.1. Marco de evaluación

Hemos escogido para nuestros experimentos un contexto de RIM *de-español-a-inglés* (consultas en español y documentos en inglés) por varias razones: (a) la conveniencia de incluir el inglés al ser la lengua dominante en la web; (b) al estar disponible en la web

[9] Empleando Google Translate para traducir de español a inglés.

[10] Realmente no estaríamos ante una *traducción* propiamente dicha desde un punto de vista lingüístico, sino sólo a efectos de recuperación, de ahí que debería hablarse más bien de *pseudo-traducción*.

más información en inglés que en otros idiomas, es lógico que actúe como *idioma destino*; (c) muchos usuarios, aún entendiendo inglés, tienen problemas para expresarse en él, por lo que emplearían su lengua materna como *idioma origen*; y (d) el empleo del español como *lengua origen* se debe a que la gran variedad de procesos morfológicos que presenta hacen de él un buen sujeto de ensayo a la hora de trabajar sobre errores ortográficos (Vilares, Otero, y Graña, 2004).

Respecto al corpus de evaluación, la colección utilizada es la *LA Times 94* (56.472 documentos, 154 MB), empleada en la *robust task* del *ad hoc track* del CLEF 2006 (CLEF, 2013), la cual reciclaba consultas de ediciones anteriores (Di Nunzio et al., 2006).[11] En cuanto a las consultas, se emplearon los 60 *topics* del *conjunto de entrenamiento* para dicha *task*,[12] que constan de tres campos: *título*, un breve título como su nombre indica; *descripción*, una somera frase de descripción; y *narrativa*, un pequeño texto especificando los criterios de relevancia. Con objeto de analizar en mayor detalle la influencia de la longitud de la consulta y la redundancia de la información que ésta contiene, se contemplaron dos series de pruebas de acuerdo a los campos del *topic* empleados para generar la consulta: (1) *consultas cortas:* empleando únicamente el campo *título* (longitud media 2,75 palabras); (2) *consultas de longitud media:* empleando *título* y *descripción* (longitud media 9,88). Por su longitud y complejidad éstas se corresponden con las consultas habituales en motores de búsqueda comerciales y otros sistemas de RI (Bendersky y Croft, 2009; Jansen, Spink, y Saracevic, 2000).

En cuanto a otros recursos empleados, los correctores requieren un diccionario del idioma, y el contextual precisa también de un corpus de entrenamiento para el etiquetador. En ambos casos se ha empleado el corpus español del MULTEXT-JOC (Véronis, 1999), con alrededor de 200.000 palabras etiquetadas y un lexicón de 15.548 términos. En el caso de las subpalabras, se empleó la versión v6 del conocido corpus paralelo EUROPARL (Koehn, 2005), con 51 millones de palabras, para la primera fase del algoritmo de alineamiento de n-gramas

[11] La otra subcolección, la *Glasgow Herald 95* no pudo ser empleada pues al haber sido introducida con posterioridad a la *LA Times 94*, no se dispone de referencias de relevancia de sus documentos (los denominados *qrels*) para gran parte de las consultas.

[12] Topics C050-C059, C070-C079, C100-C109, C120-C129, C150-159 y C180-189.

4.2. Generación de errores

Para evaluar las diversas aproximaciones introduciremos errores ortográficos en los *topics* con una *tasa de error* T creciente:

$$T \in \{0\,\%, 10\,\%, 20\,\%, \ldots, 60\,\%\}$$

donde una tasa T implica que el $T\,\%$ de las palabras incluyen errores,[13] permitiéndonos así emular incluso entornos ruidosos como aquéllos en que la entrada se obtiene de dispositivos móviles o basados en escritura a mano (ej. PDAs, bolígrafos digitales o tabletas digitalizadoras), o de interfaces por habla. Debe tenerse en cuenta que el uso de tasas tan altas no es en absoluto excesivo, pues obsérvese que en consultas cortas, como es el caso de nuestros experimentos, la tasa de error debe ser más alta para que ésta se refleje en las consultas.[14] Hemos empleado además una metodología que permite trabajar con errores humanos reales, mucho más complejos de generar y controlar, pero de mayor interés (Vilares, Vilares, y Otero, 2011).

4.3. Indexación y recuperación

Deberemos diferenciar dos casos en función de la unidad de procesamiento empleada: palabras o n-gramas de caracteres.

En el caso de usar palabras el texto es normalizado de una forma clásica empleando el *stemmer* SNOWBALL,[15] basado en Porter, y las *stopwords* de la UniNE,[16] ambos conocidos y de amplio uso, para luego ser procesado por el motor. A la hora de realizar la consulta, ésta es previamente traducida con Google Translate[17] antes de normalizarla. Consideraremos, a su vez, tres casos: (a) la consulta es traducida tal cual, con errores, siendo ésta nuestra *línea de base* (denotada *stm*); (b) la consulta es corregida previamente con el algoritmo de Savary para palabras aisladas (*Sav*), y en caso de devolver varias correcciones candidatas, se emplean todas; y (c) la consulta es corregida con el algoritmo de corrección contextual (*cont*).

[13] Donde T=0 % corresponde al *topic* original.

[14] En el caso de consultas de dos palabras se precisaría una tasa del 50 % para que haya de media un error por consulta, y con tres palabras, del 33 %.

[15] `http://snowball.tartarus.org`

[16] `http://members.unine.ch/jacques.savoy/clef/index.html`

[17] `http://translate.google.es`

En el caso de los n-gramas de caracteres (denotado *4gr*), el texto es normalizado pasándolo a minúscula y *tokenizándolo* en 4-gramas (McNamee y Mayfield, 2004b) para ser luego indexado (documentos) o traducido (consultas). En este último caso se ha empleado *log-likelihood* como medida de asociación para el cálculo de alineamientos de n-gramas, para luego, durante la traducción, reemplazar cada n-grama de la consulta original por su n-grama traducción con la medida de asociación más alta (Vilares, Oakes, y Vilares, 2007).

Nótese que no se han empleado técnicas de expansión de la consulta ni de realimentación por relevancia, y así estudiar el comportamiento de las aproximaciones consideradas sin introducir distorsiones en los resultados por la integración de otras técnicas.

El motor de indexación empleado ha sido TERRIER (Ounis et al., 2007), con un modelo de ordenación DFR InL2.[18]

5. Resultados experimentales

Los resultados obtenidos se recogen en el Cuadro 1, mostrando para cada tasa de error T: la precisión media obtenida (MAP); la caída porcentual de dicha precisión respecto a la original —T=0 %— (*%loss*), resaltando en negrita aquellos casos en los que dicha caída es estadísticamente significativa;[19] y el número de consultas, respecto al original, que han dejado de devolver documentos relevantes ($^{[\Delta\varnothing]}$). La media de ambas pérdidas se muestra al final de cada serie de resultados.

En el caso del uso de palabras como unidad de procesamiento, los resultados para nuestra *línea de base* (*stm*) muestran en todos los parámetros empleados un claro impacto negativo de los errores en el comportamiento del sistema, incluso con tasas de error bajas. Al ser mayor la importancia de cada término cuanto más corta la consulta, el impacto es mucho mayor para consultas *cortas*.

El empleo de técnicas de corrección tiene un notable efecto positivo que permite reducir dicha pérdida. En el caso del algoritmo de Savary (*Sav*), éste es muy estable en lo que respecta a la pérdida media de MAP, en torno al 24 % independientemente de la longitud (frente a 34 %/26 % para *stm*), si bien dicha caída tarda más en hacerse significativa para consultas cortas (con $T{\geq}40$ % vs. $T{\geq}30$ %). La corrección contextual (*cont*), por contra, se comporta mucho mejor con consultas más largas debido a que el reducido contexto lingüístico de las consultas más cortas limita su efectividad. De este modo en el caso de consultas cortas el algoritmo de Savary se comporta mejor (caída media del MAP de 24 % significativa con $T{\geq}40$ % vs. 29 % con $T{\geq}30$ %), mientras que con consultas medias es mejor el contextual (19 % con $T{\geq}40$ % vs. 24 % con $T{\geq}30$ %).

En el caso de los n-gramas (*4gr*), los resultados confirman su robustez también en este nuevo ámbito multilingüe, al sufrir una caída de rendimiento claramente menor que en las aproximaciones basadas en palabras, particularmente tanto para consultas cortas como para tasas de error muy altas. No sólo se muestra mucho más robusto que la *línea de base* (*stm*) para palabras (caída del MAP de 11 % vs. 34 % para consultas cortas y 14 % vs. 26 % para medias, siendo dicha caída significativa y "perdiendo" consultas sólo para los T más altos), sino que también supera a las aproximaciones basadas en corrección (caída del 11 % vs. 24 %/29 % para consultas cortas y 14 % vs. 24 %/19 % para medias). Todo ello sin aplicar ningún tipo de procesamiento específico para el tratamiento de errores.

6. Conclusiones y trabajo futuro

Se han estudiado los efectos perniciosos de los errores ortográficos en las consultas en entornos de recuperación de información multilingüe, planteándose dos posibles estrategias para abordar dicha problemática como primer paso hacia el desarrollo de sistemas de información multilingüe más robustos.

En primer lugar, una estrategia clásica basada en el uso de palabras como términos de indexación y unidad de procesamiento. En este caso se ha estudiado el empleo de mecanismos de corrección ortográfica para el tratamiento de los errores en la consulta origen, presentándose dos alternativas. Por una parte, el algoritmo de Savary, que procesa de forma aislada cada término devolviendo todas sus correcciones candidatas a distancia mínima de edición, con el consiguiente riesgo de introducir ruido si devuelve varias, siendo además incapaz de detectar errores que den lugar a palabras existentes. Por otra parte, un algoritmo de corrección contextual que

[18] Frecuencia inversa de documento con normalización 2 de Laplace.

[19] Se han empleado tests-t bilaterales sobre las MAP con α=0,05.

	stm		***Sav***		***cont***		***4gr***	
T	MAP	***%loss*** $^{[\Delta\varnothing]}$	MAP	***%loss*** $^{[\Delta\varnothing]}$	MAP	***%loss*** $^{[\Delta\varnothing]}$	MAP	***%loss*** $^{[\Delta\varnothing]}$
	CONSULTAS CORTAS							
0	0,2705	– –	–	– –	–	– –	0,1637	– –
10	0,2355	-12,94 $^{[-1]}$	0,2617	-3,25 $^{[-1]}$	0,2467	-8,80 $^{[-1]}$	0,1608	-1,77 $^{[0]}$
20	0,2153	**-20,41** $^{[-3]}$	0,2487	-8,06 $^{[-1]}$	0,2362	-12,68 $^{[-1]}$	0,1554	-5,07 $^{[0]}$
30	0,2091	**-22,70** $^{[-6]}$	0,2252	-16,75 $^{[-1]}$	0,2115	**-21,81** $^{[-1]}$	0,1542	-5,80 $^{[0]}$
40	0,1765	**-34,75** $^{[-9]}$	0,2031	**-24,92** $^{[-3]}$	0,1865	**-31,05** $^{[-3]}$	0,1455	-11,12 $^{[0]}$
50	0,1473	**-45,55** $^{[-15]}$	0,1665	**-38,45** $^{[-5]}$	0,1533	**-43,33** $^{[-6]}$	0,1409	**-13,93** $^{[-5]}$
60	0,0945	**-65,06** $^{[-22]}$	0,1360	**-49,72** $^{[-9]}$	0,1249	**-53,83** $^{[-9]}$	0,1193	**-27,12** $^{[-7]}$
media	–	-33,57 $^{[-9,33]}$	–	-23,52 $^{[-3,33]}$	–	-28,58 $^{[-3,50]}$	–	-10,80 $^{[-2,00]}$
	CONSULTAS DE LONGITUD MEDIA							
0	0,3273	– –	–	– –	–	– –	0,2042	– –
10	0,3166	-3,27 $^{[0]}$	0,3128	-4,43 $^{[0]}$	0,3147	-3,85 $^{[0]}$	0,2006	-1,76 $^{[0]}$
20	0,2952	-9,81 $^{[-2]}$	0,2825	-13,69 $^{[-1]}$	0,2917	-10,88 $^{[-1]}$	0,1800	-11,85 $^{[+1]}$
30	0,2604	**-20,44** $^{[-3]}$	0,2712	**-17,14** $^{[-2]}$	0,2890	-11,70 $^{[-2]}$	0,1782	-12,73 $^{[+1]}$
40	0,2339	**-28,54** $^{[-4]}$	0,2570	**-21,48** $^{[-2]}$	0,2655	**-18,88** $^{[-2]}$	0,1782	-12,73 $^{[+1]}$
50	0,2068	**-36,82** $^{[-4]}$	0,2141	**-34,59** $^{[-2]}$	0,2338	**-28,57** $^{[-2]}$	0,1700	-16,75 $^{[+1]}$
60	0,1500	**-54,17** $^{[-11]}$	0,1633	**-50,11** $^{[-3]}$	0,1906	**-41,77** $^{[-3]}$	0,1464	**-28,31** $^{[-2]}$
media	–	-25,51 $^{[-4,00]}$	–	-23,57 $^{[-1,67]}$	–	-19,27 $^{[-1,67]}$	–	-14,02 $^{[+0,33]}$

Cuadro 1: Resultados experimentales.

resuelve dichas limitaciones filtrando las alternativas a partir de información lingüística contextual. Nuestros experimentos demuestran que esta estrategia es muy sensible a los errores en la consulta, particularmente en el caso de consultas cortas, si bien la utilización de mecanismos de corrección permite reducir notablemente sus efectos. Asimismo durante nuestras pruebas el algoritmo de Savary se mostró más apropiado para consultas cortas, mientras que la corrección contextual fue superior para consultas de mayor longitud.

La segunda estrategia propuesta plantea el empleo de n-gramas de caracteres como unidad de procesamiento tanto para indexación como para traducción. Esto nos permite beneficiarnos de la robustez propia del procesamiento a nivel de n-grama y trabajar directamente con la consulta original con errores sin realizar ningún procesamiento a mayores. Este enfoque ha mostrado una gran robustez, con una caída del rendimiento notablemente menor que en el caso de las palabras, aún aplicando mecanismos de corrección. Se podría argumentar que el rendimiento *per se* de esta aproximación basada en n-gramas es menor que el de las aproximaciones clásicas, pero al compararlo con el caso monolingüe (Vilares, Vilares, y Otero, 2011) se observa que buena parte de tal desfase se debe a que los mecanismos de traducción a nivel de subpalabra tienen actualmente un rendimiento menor por estar aún poco desarrollados (Vilares, Oakes, y Vilares, 2007). Al mismo tiempo debe tenerse también en cuenta que se trata de un enfoque *ligero* desde el punto de vista del conocimiento y recursos empleados, y que no se basa en ningún procesamiento particular dependiente del idioma, pudiéndose aplicar para una amplia variedad de idiomas, incluso cuando la disponibilidad de información y recursos lingüísticos sea reducida. Por contra, otros enfoques más clásicos de RIM precisan recursos específicos del idioma como listas de *stopwords*, diccionarios, lematizadores, etiquetadores, corpus de entrenamiento, etc., no siempre disponibles.

De cara al futuro pretendemos mejorar los procesos de indexación-recuperación y traducción de n-gramas con objeto de aumentar el rendimiento del sistema.

Bibliografía

Bendersky, M. y W.B. Croft. 2009. Analysis of long queries in a large scale search log. En *Proc. of WSCD'09*, págs. 8–14. ACM.

CLEF. 2013. `http://www.clef-initiative.eu`.

Dale, R., H. Moisi, y H. Somers, eds. 2000. *Handbook of Natural Language Processing*. Marcel Dekker, Inc.

Di Nunzio, G.M., N. Ferro, T. Mandl, y C. Peters. 2006. CLEF 2006: Ad Hoc Track Overview. En *Working Notes of the CLEF 2006 Workshop*, págs. 21–34.

Graña, J., M.A. Alonso, y M. Vilares. 2002. A common solution for tokenization and part-of-speech tagging: One-pass Viterbi algorithm vs. iterative approaches. *LNCS*, 2448:3–10.

Graña, J., F.M. Barcala, y J. Vilares. 2002. Formal methods of tokenization for part-of-speech tagging. *LNCS*, 2276:240–249.

Guo, J., G. Xu, H. Li, y X. Cheng. 2008. A unified and discriminative model for query refinement. En *Proc. of ACM SIGIR'08*, págs. 379–386. ACM.

Jansen, B.J., A. Spink, y T. Saracevic. 2000. Real life, real users, and real needs: a study and analysis of user queries on the web. *Information Processing and Management*, 36(2):207–227.

Koehn, P. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. En *Proc. of MT Summit X*, págs. 79–86. Corpus disponible en `http://www.statmt.org/europarl/`.

Koehn, P., F.J. Och, y D. Marcu. 2003. Statistical phrase-based translation. En *Proc. of NAACL'03*, págs. 48–54. ACL.

Kukich, K. 1992. Techniques for automatically correcting words in text. *ACM Computing Surveys (CSUR)*, 24(4):377–439.

Levenshtein, V.I. 1966. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics-Doklandy*, 6:707–710.

Manning, C.D., P. Raghavan, y H. Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

McNamee, P. y J. Mayfield. 2004a. Character N-gram Tokenization for European Language Text Retrieval. *Information Retrieval*, 7(1-2):73–97.

McNamee, P. y J. Mayfield. 2004b. JHU/APL experiments in tokenization and non-word translation. *LNCS*, 3237:85–97.

Nie, J.-Y. 2010. *Cross-Language Information Retrieval*, vol. 8 de *Synthesis Lectures on Human Language Technologies*. Morgan & Claypool Publishers.

Och, F.J. y H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51. Herramienta disponible en `http://code.google.com/p/giza-pp/`.

Otero, J., J. Graña, y M. Vilares. 2007. Contextual Spelling Correction. *LNCS*, 4739:290–296.

Ounis, I., C. Lioma, C. Macdonald, y V. Plachouras. 2007. Research directions in Terrier: a search engine for advanced retrieval on the web. *Novática/UPGRADE Special Issue on Web Information Access*, 8(1):49–56. Toolkit disponible en `http://www.terrier.org`.

Rehm, G. y H. Uszkoreit, eds. 2011. META-NET White Paper Series. Springer. Disponibles en `http://www.meta-net.eu/whitepapers`.

Robertson, A.M. y P. Willett. 1998. Applications of n-grams in textual information systems. *Journal of Documentation*, 54(1):48–69.

Savary, A. 2002. Typographical nearest-neighbor search in a finite-state lexicon and its application to spelling correction. *LNCS*, 2494:251–260.

Vilares, J., M.P. Oakes, y M. Vilares. 2007. A Knowledge-Light Approach to Query Translation in Cross-Language Information Retrieval. En *Proc. of RANLP 2007*, págs. 624–630.

Vilares, M., J. Otero, y J. Graña. 2004. On asymptotic finite-state error repair. *LNCS*, 3246:271–272.

Vilares, J., M. Vilares, y J. Otero. 2011. Managing Misspelled Queries in IR Applications. *Information Processing & Management*, 47(2):263–286.

Véronis, J. 1999. Multext-Corpora. An annotated corpus for five European languages. CD-ROM. Distributed by ELRA/ELDA.