Estegoanálisis aplicado a la generación automática de estegotextos en lengua española.

Alfonso Muñoz Muñoz[1], Justo Carracedo Gallardo[1]

[1] Universidad Politécnica de Madrid, E.U.I.T Telecomunicación, DIATEL. Carretera de Valencia Km.7 – 28031. Madrid. España.
{amunoz, carracedo}@diatel.upm.es

Resumen. El presente artículo destaca las investigaciones iniciadas para analizar el potencial de la esteganografía lingüística aplicada a la lengua española, temática poco analizada para este idioma. La ocultación y detección de información enmascarada en lenguaje natural cobra una importante relevancia especialmente en entornos web 2.0 donde la información textual está más presente, si cabe, que en la "estática" red Internet, un ejemplo de ello son las redes sociales. El artículo resalta las ideas publicadas para la generación automática de estegotextos, estudiando las ventajas, inconvenientes y líneas futuras de trabajo derivadas de ellas. Se implementa una variante de un algoritmo de Peter Wayner (mimic function) en lenguaje JAVA y se analiza su utilidad esteganográfica en lengua española considerando diferentes criterios.

Palabras Claves: esteganografía lingüística, estegotextos, generación automática, CFG, NICETEXT, Peter Wayner, redes sociales.

1 Estructura del artículo.

El presente artículo está estructurado en los siguientes apartados. El segundo apartado es una breve introducción a la esteganografía y estegoanálisis, así como los trabajos previos de los autores en esta ciencia. El tercer apartado analiza el estado del arte de la esteganografía lingüística. El cuarto y quinto apartado se centra en el análisis de los mecanismos de generación automática de estegotextos. Se implementa uno de estos mecanismos y se analiza su utilidad y seguridad en lengua española. El último apartado resume los resultados obtenidos y conclusiones, avanzando posibles líneas de investigación futuras.

2 Introducción a la esteganografía. Trabajos Previos.

La esteganografía es la ciencia y el arte de ocultar una información dentro de otra, que haría la función de *tapadera o cubierta*, con la intención de que no se perciba ni siquiera la existencia de dicha información [1]. En teoría, sólo quienes conozcan

cierta información acerca de esa ocultación (un secreto) estarían en condiciones de descubrirla[1].

La ocultación de mensajes usando procedimientos esteganográficos puede tener fines legítimos o ilegítimos, que pueden ser beneficiosos para la proteger la privacidad de las comunicaciones o burlar censuras, o, por el contrario, ser vehículos para perpetrar actos criminales. Por estos motivos, en la presente década se está realizando una inversión importante en la detección de comunicaciones ocultas. El estegoanálisis es la ciencia y el arte que permite detectar esa información oculta. En general, existen dos tipos de ataques estegoanalíticos: Ataques activos y ataques pasivos. Los ataques activos se centran en la eliminación de la posible presencia de información enmascarada en un potencial estegomedio. Estas técnicas son utilizadas especialmente para atacar algoritmos de *watermarking.* Por otro lado, los ataques pasivos se centran en el estudio de los potenciales estegomedios[2] y la deducción de si almacenan información oculta. Los algoritmos estegoanalíticos más precisos (siglo XXI) son capaces no sólo de determinar el tamaño de la información oculta, sino de aplicar procedimientos de detección independientemente del conocimiento de la técnica de ocultación empleada, esto se denomina estegoanálisis a ciegas (blind steganalysis).[3] No obstante si en el proceso de ocultación se toman las medidas de protección adecuadas, para el estegoanalista es inviable (según las publicaciones actuales) conseguir la extracción y recuperación de la información real enmascarada. En general, esta tarea difícil pertenecería a la ciencia del criptoanálisis.

2.1 Trabajo previo de los autores.

En 2007, como consecuencia del diseño de una arquitectura genérica de detección de información esteganográfica (2005), se publico la herramienta libre Stegsecret (http://stegsecret.sourceforge.net) que demostraba que era viable la detección de información ocultada con muchos de los programas esteganográficos disponibles en Internet (camouflage v1.2.1, inThePicture v2, JPEGXv2.1.1, técnicas EOF, etc.), así como se implementó algunas de las técnicas de estegoanálisis "genéricas" más difundidas hasta la fecha (ataques visuales, test chi-square y RS-attack). Esta investigación permitió observar los umbrales mínimos de precisión en la detección de algunos de los algoritmos de estegoanálisis más famosos. Es decir, si se oculta, debidamente, una cantidad mínima de bits los algoritmos de detección no son capaces de determinar si en un estegomedio dado hay información oculta o no. Este umbral depende de varios factores pero suele estar en torno a decenas de octetos. La idea es clara al ocultar menos información (si se utilizan técnicas que modifiquen el medio) el impacto será menor. En la actualidad, para conseguir esto se recurre al uso de

---

1 En criptografía no se oculta la existencia del mensaje sino que se hace ilegible para quien no esté al tanto de un determinado secreto (la clave). Por este motivo, los mensajes que se procuran ocultar usando técnicas esteganográficas, habitualmente, son previamente cifrados.

2 Se denomina estegomedio a la cubierta o medio original que se utiliza para ocultar una información.

3 El concepto de estegoanálisis a ciegas aprovecha el uso de diversos clasificadores (SVM, Fisher, etc) que mediante la definición de características propias de cada medio consiguen diferenciar entre cubiertas originales y potenciales estegomedios.

matrices de codificación[4] y distribución de la información en varios portadores. Este último aspecto es especialmente interesante en las nuevas formas de comunicación colaborativa, donde puede dificultarse la detección de información esteganográfica mediante la distribución de información multiproveedor y multiportador. En el caso de las redes sociales, a estos factores se le suma la dificultad de automatizar el estegoanálisis de los datos publicados al existir mecanismos como: CAPTCHAS que dificultan el trabajo de software de rastreo, la existencia de grupos cerrados de usuarios, etc. La distribución multiproveedor y multiportador dificultaría la tarea si un proveedor de un servicio estuviera interesado en analizar la información almacenada por él. En este sentido se publicó en 2008 los artículos: *Herramienta DCST. Automatización de estegoanálisis en Redes Sociales y Detection of distributed steganographic information in social networks.* Para profundizar en algunos de los temas tratados véase [2].

La combinación interesante de esteganografía y redes colaborativas (por ejemplo, redes sociales, p2p o p2m) dirige nuestro interés actual. Entre las líneas de investigación los esfuerzos se centran en la ocultación y detección de información oculta en información textual en lengua española, aplicada especialmente a entornos colaborativos, para ello se está trabajando desde un punto de vista multidisciplinar, con especialistas en esteganografía y lingüistas con dominio de lengua española.

3 Estado del Arte. Esteganografía Lingüística.

Una de las técnicas más antiguas de ocultación de información, y posiblemente una excelente opción en determinadas situaciones en la actualidad, consiste en enmascarar información utilizando como estegomedio el lenguaje natural. La información textual, está presente en todo, Internet y las redes sociales es una buena muestra de ello. Este hecho hace que su utilización como estegomedio dificulte enormemente la capacidad de un estegoanalista en "separar el grano de la paja"[5].

Las técnicas de ocultación basadas en información textual, en general, se basan en la utilización de textos existentes (se modifican) o la creación de textos de forma automática ("a medida"). La seguridad de estos estegotextos debe ser analizada desde diferentes puntos de vista, considerando ataques lingüísticos (sintáctico, semántico y de coherencia), por parte de máquinas y analistas, y ataques puramente estegonalíticos y estadísticos (análisis de entropía[6], análisis de frecuencia de caracteres-palabras, ataques basados en conocimiento de cubierta original y cubierta modificada, etc). Estos últimos ataques pueden ser minimizados, en general para cualquier estegomedio, con algunos de los procedimientos ya comentados (matrices de codificación y distribución) u otros que compensen las perturbaciones introducidas en

[4] El concepto de matriz de codificación hace referencia a procedimientos matemáticos que mejoran la relación información insertada-modificación de un estegomedio.

[5] Algunas herramientas que ocultan información en texto son: *Nicetext,TextHide, TextSign, StegParty, Texto, Spammimic, Stegano, Steganosaurus, Mimicry Applet, Tyrannosaurus lex, Snowdrop*

[6] Estudios mediante la ecuación de "entropía de Shannon" para detectar datos aleatorios (información cifrada) u otras alternativas que aproximan el valor pero son más eficientes en el procesamiento de datos, como la de Shamir y Van Someren [4].

un medio. También es útil la aplicación de ideas, más interesantes, como puede ser la negación plausible, como se verá en los siguientes apartados. En general, técnicas que necesitan una gran cantidad de información textual para ocultar un cantidad de información no muy abultada. Entre otras, porque el lenguaje natural, como estegomedio, es muy poco redundante (ruidoso) en comparación con estegomedios como son las imágenes o videos, lo cual hace más complicado la creación de algoritmos robustos de ocultación de información en lenguaje natural.

A lo largo de los siglos, no obstante, se han documentado múltiples formas de ocultación en: cartas, libros, telegramas, poesías, canciones, revistas, periódicos (por ejemplo, *newspaper code* en la época Victoriana o la verja de Cardano en el siglo XVI), en canales de mensajería instantánea (messenger, IRC), basado en el "ruido" de las traducciones automáticas, basados en lenguajes de marcado (HTML y XML), etc [2]. Es común ver su clasificación en términos de *open codes*[7] *y semagrams*[8], en terminología inglesa.

La década de los 90 del siglo pasado supuso el renacimiento de estos antiguos procedimientos gracias a la publicación de Peter Wayner en 1992 de las funciones mimic y otra serie de trabajos sobre el uso seguro de gramáticas en la generación de estegotextos [3]. Estas ideas ha evolucionado en la presente década y en la actualidad diferentes resultados esteganográficos en lenguaje natural se han obtenido para idiomas tan dispares como: inglés, ruso, mandarín, coreano, persa, etc. A pesar de esto, no existen estudios avanzados de esteganografía lingüística en lengua española, hasta lo que tenemos constancia. De la información consultada únicamente es destacable el artículo "*Using Selectional Preferences for Extending a Synonymous Paraphrasing Method in Steganography*" del mejicano Hiram Calvo y el ruso Igor. A. Bolshakov (2004) [3], aplicado a fragmentos en lengua castellana. En cualquier caso si se desea profundizar en esta temática para lengua española debe considerarse, en la actualidad, las siguientes líneas de investigación para ocultar información en lenguaje natural (textos):

1. **Generación automática de estegotextos.** Se analiza en detalle en los siguientes apartados.

2. Generación de estegotextos basados en la modificación de textos existentes. El mecanismo más socorrido, a lo largo de la historia, para ocultar información en lenguaje natural consiste en la modificación de textos o documentos existentes mediante diferentes procedimientos. El problema principal de estos mecanismos reside en la propia modificación, ya que si el texto original utilizado, para la ocultación, fuera accesible por un potencial estegoanalista, este hecho facilitaría un ataque clásico de comparación "texto en claro – texto codificado" que delataría la presencia de información oculta. La otra opción sería generar manualmente el texto original (o automáticamente), que no debería ser accesible, cuya modificación esteganográfica se distribuiría en el canal deseado, por ejemplo,

---

[7] Los open codes genéricamente se refieren a textos de apariencia inocente, que ocultan información recuperable utilizando ciertas letras, palabras, frases del texto o comunicación. Métodos basados en esto son: Cues, Null Ciphers, Jargon Code, Grilles, etc [2].

[8] Tipo de técnicas que consisten en la utilización (variación) de la estructura y formato de los elementos de un texto, aunque visibles, no por ello son fáciles de detectar.

mediante un artículo de opinión en un blog. En cualquier caso, cuando se desea modificar el contenido de un texto/documento con fines esteganográficos o de marcado digital debe considerarse:

a) Modificaciones léxicas. Estos procedimientos consisten en la ocultación de información mediante la sustitución/modificación de palabras. El método más analizado es la sustitución basada en el uso de sinónimos. Desde que esta idea fuera trabajada por Chapman y Davida en 1997 es considerada como una excelente opción y estudiada en diversos lenguajes [3]. El mayor problema con esta técnica es que, o no existen, o son muy pocos los sinónimos puros en una lengua, es decir, dos palabras que signifiquen exactamente lo mismo en cualquier contexto. Por este motivo, conseguir herramientas prácticas con estos principios (ya sea para ocultación de información en general o para marcado digital de textos) requiere de sofisticados mecanismos para determinar cual es la ambigüedad de una palabra dada en un contexto determinado, para saber si puede ser reemplazada o no por otra palabra. Para ello se requieren analizadores WSD[9] robustos y estudios estadísticos que indiquen de los sinónimos disponibles para una palabra cuales son los más aconsejados.

b) Modificaciones Sintácticas y Semánticas. Los algoritmos de ocultación más robustos basados en esteganografía lingüística (se intuye) deberían ser capaces de aplicar modificaciones sintáctico-semánticas a un texto para ocultar información sin perder la coherencia y la semántica del texto. Esta investigación está completamente abierta en su aplicación a diferentes lenguas (algunos de nuestros estudios para lengua española van en esta dirección). Algunos recursos sintácticos documentados en diferentes lenguas (inglés, chino, coreano, turco, ruso, persa, etc) para ocultar información son: intercambio de voz activa/pasiva, desplazamiento en las posiciones de adverbios, orden en términos unidos por conjunciones (por ejemplo, listo y guapo o guapo y listo), etc [3]. Por otro lado, se sigue trabajando en mecanismos que aprovechándose de descripciones semánticas (ontologías aplicadas a la esteganografía) faciliten la ocultación de información considerando la semántica y coherencia de un texto. Un ejemplo sencillo consiste en la inserción de sentencias/términos semánticamente "vacíos" (no afectan al contexto), por ejemplo, en inglés la inserción de términos delante del sujeto de una sentencia como: *Basically, it seems that*, etc [3].

e) Traducciones a otros idiomas. La idea de estos procedimientos consiste en ocultar información basándose en la posibilidad de traducir una sentencia, de un lenguaje concreto, en varias sentencias "equivalentes" en un lenguaje destino, entre las cuales se puede elegir estableciendo un sistema binario de ocultación de información [3].

[9] En lingüística computacional WSD (word sense disambiguation) hace referencia al proceso de identificar el sentido de una palabra en una frase o contexto determinado cuando dicha palabra tiene más de un significado.

f) Errores tipográficos y ortográficos. Abreviaturas y símbolos de puntuación. Estos mecanismos pueden parecer triviales desde un punto de vista lingüístico, sin embargo, pueden presentar utilidad esteganográfica si los textos creados con estas modificaciones se insertan en "canales" donde este tipo de errores sean frecuentes. Por ejemplo, incluir textos esteganográficos basados en faltas ortográficas en foros en Internet donde los textos escritos presentan muchos errores de este tipo (lo cual, por desgracia, es común). De la misma forma, se han documentado diferentes procedimientos para ocultación de información utilizando abreviaturas de palabras, por ejemplo, de manera ingeniosa en mensajes sms/mms [5]. En esta línea, los símbolos de puntuación (punto, coma, punto y coma, etc), más exactamente su colocación o no en zonas de un texto, también pueden ser utilizados para establecer sistemas binarios de ocultación. La creación de reglas generales para aplicar esta idea a los textos de un lenguaje dado no es nada sencillo. Por ejemplo, en turco una coma puede ser usada después de un sujeto si esta está relativamente distante del verbo. La inconsistencia de esta definición hace que su automatización sea difícil. [3].

g) Ocultación basada en formato. Este es el mecanismo más tradicional para ocultar información basado en el formato-estructura de un texto. Los recursos más utilizados son: uso de caracteres invisibles, separación entre líneas o palabras (por ejemplo, uso de espacios de tamaño variable entre palabras[10]) y codificación de información basada en sucesivos cambios del formato del texto (estilo de fuente, color, tamaño de letra, subrayado, negrita, cursiva, mayúsculas, etc) [3]. Estos mecanismos pueden favorecer la ocultación de una cantidad razonable de información pero no están exentos de problemas. A menudo, simples ataques activos (que anulan el formato) eliminan la información oculta. No obstante, en los últimos dos años, especialmente por el interés la comunidad científica china en el estegoanálisis de estegotextos, se han publicado estudios de interés. Véase por ejemplo [6][7].

En los siguientes apartados, dado que existe poca documentación al respecto, se analiza las ventajas e inconvenientes de alguno de los procedimientos de generación automática de estegotextos más conocidos en su aplicación a lengua española.

4 Generación automática de textos. Context-Free Grammar.

La generación automática de estegotextos permite la creación de textos originales que enmascaran una información oculta. Su ventaja fundamental reside en la posibilidad de crear estegotextos únicos para cada comunicación, de forma que se dificulte ataques basados en comparaciones texto original-estegotexto. En las últimas décadas los esfuerzos en este sentido se centran, principalmente, en la generación de estegotextos que imiten la gramática (sintaxis) y la estadística de un texto "típico" en una lengua concreta.

En la década de los 60 el excepcional lingüista A. Noam Chomsky postulo la gramática generativa. Esta gramática se definió como el conjunto de reglas innatas

10 El tamaño en dpi (dots peer inch) que separa palabras o líneas en un texto puede ser configurado con sistemas tipográficos avanzados, como por ejemplo, TEX/LATEX.

que permite traducir combinaciones de ideas a combinaciones de palabras y en este sentido, **la gramática se convertía en un sistema combinatorio discreto que permite construir infinitas frases a partir de un número finito de elementos** mediante reglas diversas que pueden formalizarse mediante una gramática formal gobernada por normas de transformación. Según esta teoría de lenguaje formal una CFG (Context-Free Grammar) se define como una gramática en la que cada regla de producción es de la forma v::=w, donde v es una variable y w es una cadena de símbolos no terminales y/o terminales [3]. En la década de los 90 Wayner [3] insistió en la posibilidad de utilizar estas construcciones (CFGs) para la generación automática de estegotextos, ya que implícitamente se generarían textos, que al menos, tendrían validez gramatical-sintáctica[11].

Variable_Inicio S ::= *AB (.5)* | *AC (.5)*
A ::= “Buenos Días”(.25)|“Buenas Tardes”(.25)|“Buenas noches” (.25)|“Hola”(.25)
B ::= “estimado amigo” C (.5) | “estimado compañero” C (.5)
C ::= “Juan” D (.25) | “Pedro” D (.25) | “Lucas” D (.25) | “Tomas” D (.25)
D ::= “quedamos algún día para” E (.5) | “dame tu número de teléfono para” E (.5)
E ::= “hablar” F (.5) | “charlar” F (.5)
F ::= Un saludo (1.0).

Fig. 1. Ejemplo de PCFG en lengua española en formato BNF. Ocultación máxima de 7 bits.

La ocultación de información se realiza mediante la selección de elementos concretos dentro de una regla específica, regla que es elegida mediante algún algoritmo de selección concreto. Aunque Wayner se esforzó en formalizar la construcción de CFGs seguras con utilidad esteganográfica y analizar su seguridad, es cierto que, su utilidad esteganográfica debe ser muy matizada. El primer problema, que parece insalvable, es que la gramática debe permanecer privada (emisor y receptor la deben conocer), ya que si no es así un atacante podría inferir fácilmente la información oculta. Este problema es mayor si la gramática es estática-manual y es comprometida, lo cual requiere un nuevo proceso tedioso y costoso de generación de la misma. La calidad del estegotexto depende claramente de la gramática, y si esta es estática-manual (pocas reglas) es más que probable la repetición de frases y términos en el estegotexto, facilitando a los estegoanalistas su trabajo. Aunque la gramática sea generada automáticamente de uno o más textos de referencia, conocidos por emisor y receptor, debe considerarse otros análisis al generar algoritmos esteganográficos basados en CFGs: a) Las palabras (términos) en una CFG se relacionan con sus vecinos en formas fijas. Aunque se añadan modelos estadísticos a las gramáticas (probabilistic context-free-grammars –PCFG-), para dificultar ataques de análisis, siempre existirán correlaciones mutuas si se quiere que el texto sea legible por un humano [3], b) ataques basados en estudio de terminales (información última de cada regla) ya que aunque las variaciones de texto creados puedan crecer sustancialmente con el tamaño de una gramática dada, el número de terminales está limitado por el tamaño de la gramática, lo cual significa que forzosamente, si el texto es lo

[11] Las CFGs han jugado un papel nuclear en el diseño de lenguajes de programación y compiladores, así como en el análisis de la sintaxis del lenguaje natural. Una CFG se compone de terminales, variables y producciones.

suficientemente grande, combinaciones lineales de terminales se tienen que producir y por tanto repetir.

En la práctica resulta realmente complejo utilizar CFGs en herramientas públicas de manera robusta en la concepción actual. Un intento notorio, de los pocos destacables, fue el sistema NICETEXT, del que se pueden extraer ideas para nuevos diseños. Esta herramienta introdujo un sistema de generación automática de estegotextos [3] basado en PCFGs, que se pueden crear dinámicamente, y un procedimiento de sustitución basada en palabras categorizadas por contenido semántico. Su funcionamiento es sencillo: $NICETEXT_{D,S}(C) \rightarrow T$ y $SCRAMBLE_{D}(T) \rightarrow C$, siendo D un diccionario de palabras categorizadas por tipo y S (meta-style source) un conjunto de reglas de estilo de escritura basadas en las ideas de gramáticas libres de contexto. Estas gramáticas se eligen independientemente de la información a ocultar (basado en criterios estadísticos) y la información se oculta mediante la selección de las palabras concretas que correspondan a un término concreto de la regla definida. Entre las ventajas destacables de esta herramienta, se encuentra que el receptor no necesita la gramática para recuperar la información enmascarada, lo cual permite utilizar en caso extremo una gramática única por comunicación, de la riqueza que se desee, y que en el diccionario creado se utilizan palabras categorizadas semánticamente (extraída de algunas fuentes de texto y clasificada mediante analizadores morfológicos), lo que facilita la creación de fragmentos de textos con un cierto nivel semántico. A esto, se le une la posibilidad de utilizar principios de negación plausible aplicada a esteganografía lingüística [3]. En NICETEXT puede suceder que la información a ocultar no sea suficiente para completar todos los términos de una regla seleccionada de la gramática, por ello se necesita utilizar una información aleatoria para seleccionar el resto de palabras hasta completar la regla en curso. Como esta selección es intrínseca al sistema, se podría utilizar a su vez para ocultar información cifrada, de forma, que un analista no podría determinar si el texto generado de esa información aleatoria se debe al proceso natural del sistema (es plausible) o a otra información enmascarada (negación plausible).

Independientemente de las ventajas conceptuales aportadas y obviando defectos (sustituciones no válidas en contexto, anomalías entre el estilo de escritura seleccionado y el vocabulario empleado, etc) en 2008 la comunidad científica china publicó, una vez más, una serie avances en estegoanálisis lingüístico, en concreto atacando a herramientas como NICETEXT, TEXTO o basadas en cadenas de Markov, que demuestra las múltiples consideraciones a tener en cuenta. Sus ataques se basaron en la suposición que en un texto natural las palabras se distribuyen de manera no equitativa, es decir, algunas palabras se repiten frecuentemente en algunos lugares pero rara vez en otros. Basándose en esto calcularon una serie de estimadores basados en las localizaciones de las palabras en un texto bajo estudio (estimadores que luego entrenarían un Support Vector Machine). Sus resultados indican, a falta de ser contrastados con otros estudios independientes, que el ratio de detección excede del 90% para estegotextos de tamaño en torno a 5KB [10].

En resumen, las CFGs (en general, PCFGs) podrían ser utilizadas, al igual que en otros lenguajes, en lengua española (algunos ejemplos sencillos se han documentado [11]) con fines esteganográficos, pero dependiendo del volumen de información a ocultar (en torno a miles de octetos) no de un modo seguro en su concepción actual, lo cual tiraría por tierra el esfuerzo (costoso) de generar gramáticas de calidad. A

menudo, se teme que los estegotextos resultantes sufran de ataques que analicen la falta de semántica y retorica global de un estegotexto (o de sus fragmentos) que alerte la presencia de información oculta. Bien es cierto que en la actualidad las CFGs o PCFGs aplicadas a esteganografía deben ser muy depuradas, a un nivel más inicial aún, deben resistir ataques estadísticos basados en frecuencia de palabras, repetición y localizaciones de términos en un texto. Es conjeturable que en un futuro cercano estas ideas de generación automática de textos puedan ser mejoradas con los avances actuales en diferentes lenguas, por ejemplo en inglés, en sustituciones léxicas y transformaciones semánticas. Esta línea de investigación, que está en curso para lengua española, proporcionará sistemas más seguros a costa, se supone, de una menor capacidad de ocultación, frente a ataques estadísticos, sintácticos y, ahora sí, semánticos.

5 Análisis del algoritmo de imitado de Peter Wayner en lengua española. Herramienta Stelin.

Peter Wayner en la década de los 90 publicó un procedimiento de generación automática de estegotextos (T) basado en el imitado estadístico de una o más fuentes de textos (S) que es interesante analizar [3]. La idea es sencilla: Cójase una función de imitado f que modifique un fichero A de forma que asuma las propiedades estadísticas de otro fichero B. Es decir, si p(t,A) es la probabilidad de que una cadena t suceda en A, entonces una función de imitado f, hace que la p(t,f(A)) sea aproximadamente p(t,B) para toda cadena t de tamaño menor que n. La complejidad del modelo estadístico de imitado (análisis de frecuencia) depende, precisamente del orden estadístico n (orden de complejidad del algoritmo). Según está idea, Wayner definió el siguiente algoritmo de imitado:

1. Constrúyase una lista de todas las diferentes combinaciones de n letras que ocurran en S y contabilizar el número de veces que ocurren en S.
2. Elegir una de ellas aleatoriamente que actuará de semilla inicial. Esto generará las primeras n letras de T (el estegotexto).
3. Repetir este punto hasta que se genere todo el texto deseado
    a. Cójase las n-1 letras siguientes de T
    b. Buscar en la tabla estadística (creada) todas las combinaciones de letras que comienzan con esas n-1 letras.
    c. La última letra de esas combinaciones forma el conjunto de posibles elecciones para la siguiente letra que será añadida a T.
    d. Elegir entre esas letras y usar la frecuencia de sus ocurrencias en S para “evaluar” cuál es la mejor elección.
    e. Añadirla a T.

Por ejemplo, un primer orden de imitado genera caracteres aleatorios de acuerdo a su distribución estadística. En un segundo orden imita la distribución de parejas de caracteres de los textos S de entrenamiento, y así continuamente para mayor orden. El proceso de ocultación de información se realiza mediante la selección de las opciones

de la próxima letra a mostrar. Wayner justificó como esto se podría hacer, entre otras opciones, utilizando un árbol de Huffman, que basándose en las frecuencias de aparición de los caracteres (por ejemplo) les asignaría un código (código que se utilizará para ocultar una información). Si la selección de las ramas de este árbol (que imita la estadística a la fuente), es aleatoria el texto resultante imitará (o se aproximará) a la distribución estadística del texto fuente. Se supone (por la información publicada [3]) que para lengua inglesa, dependiendo del texto y del orden (texto de al menos decenas de KB y orden mayor que 8), pueden obtenerse algunos estegotextos con validez léxica y sintáctica, e incluso con apariencia semántica-estructural. En lengua española, por las pruebas iniciales realizadas, esta afirmación es difícil de mantener. La ocultación de unas pocas decenas de octetos producirá estegotextos con algún error léxico o gramatical. Entre otros motivos, porque la consideración del carácter como unidad atómica de entrenamiento para generar estegotextos puede producir léxico no válido en el contexto textual y esto no se soluciona al considerar textos más grandes (más posibilidades de elección) y orden de complejidad mayor (lo que implica un factor de expansión del estegotexto mayor).

"que no cese la producción y sus políticos. La seguridad y soberanía y la seguridad y soberanía alimentaria desde una posición de defensa campesinas de los poderes económicos, sociales, económicas y culturales. E) El rechazo social a las políticas de producir riqueza, el capitalista de la soberanía y la seguridad alimentaria en el mercado mundial, así como su reconocimiento recíproco como sujetos de derechos de la pobreza y la exclusión. Hoy no se producen los alimentos), homogeneizando culturas, criterios y técnicas productiva y culturales, manteniendo el control de los poderosos, el incremento de la des"

Fig. 2. Ocultación de 10 octetos de información (0xF0, 0x0F, 0xAA, 0x71, 0xF0, 0x0F, 0xAA, 0x71, 0xF0, 0x53) utilizando un texto[12] de 24KB y orden de complejidad 11. Expansión 1:61

La idea de Peter Wayner podría ser mejorada (o eso se piensa) si se considera un nivel de atomicidad de entrenamiento diferente, por ejemplo, la utilización de la palabra en lugar del carácter, dado que se espera que los resultados mejoren, al menos sintácticamente. Basado en estos criterios y en algunas mejoras adicionales, se implementada una herramienta de código libre para la generación automática de estegotextos. La herramienta Stelin, desarrollada en lenguaje JAVA, facilita el análisis del algoritmo de Peter Wayner y la variante propuesta en lengua española (http://stelin.sourceforge.net).

El algoritmo implementado en Stelin para imitar una fuente de texto de entrenamiento, considerando como nivel de atomicidad la palabra[13], es el siguiente:

1. El proceso de generación se basa en el análisis de bloques de n

[12] El texto utilizado en el ejemplo es el artículo titulado: "Seguridad Alimentaria y sus condiciones de posibilidad". Descargable de: http://www.kaosenlared.net/noticia/seguridadalimentaria-condiciones-posibilidad.

[13] Otro nivel atómico sería posible: un párrafo, verso en un poema, etc. Esto condicionaría, en general, el factor de expansión (mayor) y la capacidad de ocultación (menor).

palabras extraídas del texto de entrenamiento mediante una ventana deslizante que se desplaza una posición para cada nuevo bloque. Es decir, el primer bloque tendrá los términos de 0 a n-1, el segundo bloque de 1 a n, y así sucesivamente.

2. N define el orden de complejidad del algoritmo. Lo que significa el número de palabras a considerar consecutivamente y por tanto, su aparición viene condicionada por la aparición de palabras que le preceden o suceden.

3. Las palabras se relacionan mediante nodos enlazados, en los que se contabiliza el número de veces que se han repetido en el texto de entrenamiento. Según esto, existirá una tabla raíz que almacenará todas las "palabras diferentes" que existan en el texto fuente.

Basado en los anterior, el algoritmo de generación de estegotextos, funcionaría, en general, de la siguiente manera:

a) Se selecciona una "palabra" aleatoriamente de la tabla raíz (otro criterio podría considerarse, con fines sintácticos, por ejemplo, hacer que el texto empezase por un artículo), de esta forma para un mismo texto de entrenamiento se podrían obtener diferentes estegotextos.

b) Si esta palabra no tiene sucesores (no apunta a otro nodo), se elije otro término de la tabla raíz (paso a). Si el nodo sucesor solo tiene una palabra, esta palabra se añade al estegotexto (no es posible ocultar información en este caso) y se elige el siguiente nodo disponible. Si el nodo sucesor tiene varias palabras posibles entre las que elegir se elije aquella cuya rama del árbol de Huffman, generado de las posibles palabras (y sus frecuencias), coincida con la información a ocultar, y se elige el siguiente nodo disponible.

c) Si se llega al último nodo (orden n=8, por ejemplo, 8 palabras consecutivas) se elige la última palabra seleccionada para el estegotexto y se vuelve al paso b). Este proceso se repite hasta que se genere el estegotexto que oculta la información deseada.

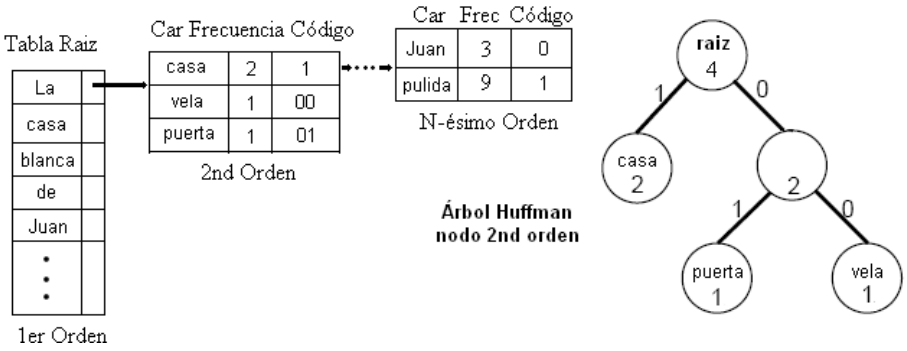

Fig. 3. Variante del algoritmo de P. Wayner . Nivel de Atomicidad = Palabra.

Esta variante, genera estegotextos de mayor tamaño (a mayor nivel de atomicidad es más probable que los elementos no tengan tantos diferentes sucesores), pero es más fácil obtener textos con validez léxica y sintáctica (no exentos de pequeños fallos), e incluso, en ocasiones, con apariencia semántica.

[INICIO TEXTO] *pesadilla. Está el sol en el ocaso. Suena el agua clara no mitiga, la amargura del tiempo de mentira, de infamia. A España toda, la luna llena, el ojo encandilado del búho insomne sueño mío! ¡Este frío de un amanecer en la tierra, y en este nuevo ejido sin duda, el amor a mujer el que llevó a un límite infranqueable la desubjetivación del sujeto. "¿Y cómo no intentar —dice Martín— devolver a mi oído, por la ventana de mi estancia, iluminada por esta luz invernal, —la tarde gris de plomo y azul de plata, con manchas de roja herrumbre, todo envuelto en luz violada. ¡Oh tierras de Alvargonzález, en el corazón de una tarde inmensa; mas falta el hilo entre los dos! Al borrarse la nieve, se alejaron los montes de la sierra. La tarde está cayendo frente a los caserones de sus lares; la tempestad llevarse los limos de una manera española, que fue casarse con una tarde clara y amplia como el hastío, cuando el eje del planeta se vence hacia el poeta admira y calla, el sabio mira y la noche azul ardía toda sembrada de estrellas. ¡Padre!, gritaron; al fondo de la laguna serena cayeron, y el eco ¡padre! repitió de peña denegrida, vuelve mi corazón a su faena, con el viento... ¡el viento de la tarde en su* **[FIN TEXTO]**

Fig. 4. Ocultación de 16 octetos (0xF0, 0x0F, 0xAA, 0x71, 0xF0, 0x0F, 0xAA, 0x71, 0xF0, 0x0F, 0xAA, 0x71, 0xF0, 0x0F, 0xAA, 0x71). Texto Fuente obra “Poesías Completas” de Antonio Machado (290KB texto plano) y orden de complejidad 9. Expansión 1:76.

En los textos mostrados en la Fig.2 y Fig.4, los estegotextos no finalizan necesariamente con una estructura puramente sintáctica. Esto puede solucionarse, por ejemplo, mostrando términos (ocultando una información aleatoria de relleno) hasta encontrar un fin de cadena (por ejemplo, un punto). Este hecho facilitaría a su vez la ocultación de una pequeña cantidad de información basado en principios de negación plausible. En cualquier caso, la selección de los textos de entrenamiento y orden de complejidad son vitales para la generación de estegotextos de calidad. Diferentes tipos de textos podrían ser considerados como fuente para ocultar información (poemas, novelas, artículos periodísticos, código de programación, etc). Si el texto fuente es más grande es más probable que existan diferentes alternativas que sucedan a una palabra y por tanto la capacidad de ocultación sea mayor (piénsese en lengua española por ejemplo en la presencia de preposiciones). Desde un punto de vista lingüístico debería, al menos, evitarse o filtrase fragmentos de texto que claramente afecten a la coherencia en los estegotextos creados. Entre estos, índices, títulos, numeraciones (a), b), c), I, II, III), fechas, referencias, etc. Por otro lado, a falta de una mejor formalización, las pruebas realizadas indican, que en general, un orden 8 o superior proporciona unos resultados léxicos y sintácticos razonables. La cuestión está en determinar, si el esfuerzo merece la pena, si para un texto dado un orden N+1 es mejor (estadística y lingüísticamente) que un orden N o menos, ya que a mayor orden, es más probable, que el factor de expansión sea mayor y estegotexto generado (más grande) sea más “atacable”. En cualquier caso, por las pruebas realizadas este procedimiento sólo permitiría la ocultación de una breve cantidad de información (decenas de octetos) de forma segura (siempre podrían distribuirse pequeñas informaciones en diferentes estegotextos). Esta cantidad de octetos permitiría el intercambio de breves mensajes de información, urls o claves criptográficas. Por ejemplo, una información de 16 octetos (128 bits) podría codificar, mediante un

alfabeto de 32 elementos (27 letras y 5 símbolos adicionales), hasta 25 letras. Por ejemplo, un mensaje de movilización como: “manifa ramblas a las ocho”.

5.1 Problemas estadísticos del algoritmo de Peter Wayner y variantes.

La seguridad de esta propuesta, como la de otras, debe ser analizada desde diferentes puntos de vista, no exclusivamente ataques lingüísticos. Un ejemplo de ello son los estudios estadísticos. En la práctica la aproximación estadística de la fuente de entrenamiento (texto) realizada por la idea de Wayner y variantes dependerá de varios factores, entre otros de la función de imitado utilizada. Por ejemplo, una posible codificación de tres caracteres (a,b,c) con probabilidades (0.89,0.07,0.04) utilizando el algoritmo Huffman sería (0,10,11). Si su función inversa es usada como función de imitado los caracteres aparecerían con frecuencia (0.5,0.25,0.25) lo cual dista de ser una aproximación estadística razonable. Esto es debido a que al utilizar un árbol binario para representar las opciones de los caracteres (u otro nivel de atomicidad) su distribución estadística de los mismos siempre será una potencia negativa de 2. Esta potencia depende de la distancia entre la raíz y la hoja correspondiente del árbol. Si existen muchos caracteres en el árbol su profundidad será mayor y la aproximación también (se han propuesto otras alternativas para aproximar mejor los valores de la fuente [3]). En general, utilizar un orden de complejidad mayor mejorará la aproximación estadística de la fuente. El límite del orden o del tamaño del texto fuente a procesar viene dado por los recursos hardware involucrados en las operaciones. Por este motivo, en la herramienta Stelin, en el caso de atomicidad basada en palabra, se implementan diferentes modos de actuación con el orden de complejidad, ya que para textos fuentes grandes es posible que el algoritmo desborde en memoria. Por ello, se puede configurar el orden concreto de almacenamiento de términos en cada nivel de recursividad del algoritmo, limitar el orden general de cada nivel a un valor fijo, etc.

La utilización de un nivel de atomicidad mayor que el carácter no tendría porque cumplir las aproximaciones estadísticas justificadas. Lo cierto es, que por los estudios en curso, al menos a nivel de carácter la variante implementada se aproxima a la distribución reflejada en la fuente de entrenamiento para órdenes grandes, 8 o más. (véase por ejemplo, Fig.5). No obstante, aparte de otros ataques, está por ver si estas propuestas son seguras frente a ataques basados en localización no equitativa de términos en un estegotexto generado [5]. En este caso concreto, por la información publicada, la propuesta estegoanalítica tendría cierto éxito en análisis sólo si los estegotextos son de al menos de unas cuantos miles de octetos, caso que no se da, por ejemplo, en Fig.2 y Fig.4.

Entre otros ataques sería interesante analizar qué sucedería si un atacante conociera el texto fuente de entrenamiento (que es secreto y compartido entre emisor y receptor) y el orden de complejidad utilizado para generar un estegotexto concreto. Si esto fuera así el atacante podría, reconstruyendo los árboles de Huffman correspondientes, decodificar cada palabra del estegotexto a un código concreto. Información binaria que podría ser analizada para descubrir la información enmascarada u observar la posibilidad de la existencia de una información cifrada (alta entropía). Por si esto sucediera, en primer lugar, la información a ocultar debería ser cifrada (en la

herramienta Stelin se utiliza un cifrador basado en el algoritmo AES-256 en modo contador -CTR mode). Además de esto, los ataques derivados de análisis de entropía y recuperación de información podrían dificultarse de diversas maneras, algunas de ellas comentadas ya en el artículo. En el caso concreto de la herramienta Stelin este hecho se dificulta aplicando ideas clásicas de cifradores basados en reducción de redundancia [12]. Stelin utiliza un generador PRNG (AES-256 en modo contador) que asigna a cada rama de cada árbol de Huffman (0 o 1) de forma aleatoria en función de una clave (no de forma fija a la rama derecha 0 o 1 y a la izquierda 1 o 0)

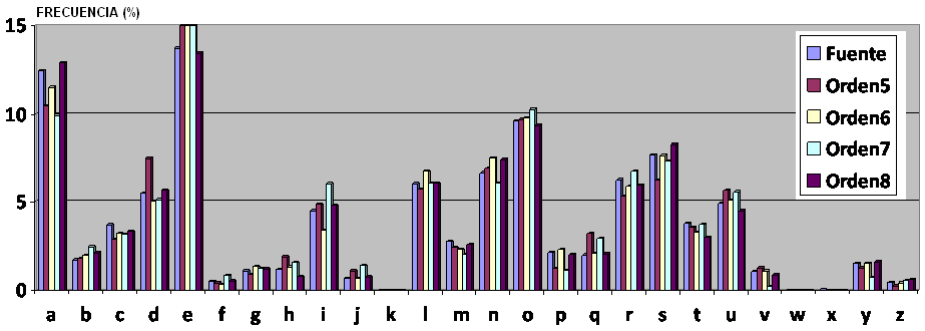

Fig. 5. Ejemplo de comparación de la distribución de frecuencias de caracteres de una fuente de texto de entrenamiento (los 13 primeros capítulos del Quijote con un total de 167.313 caracteres) y los estegotextos generados de ocultar 256 octetos (2048 bits) de información oculta a partir de dicha fuente (como nivel de atomicidad la palabra).

de forma que un atacante que conozca el texto de entrenamiento y el orden tendría dificultades en asignar un código concreto a una palabra del estegotexto determinado. Esto dificultaría, entre otras cosas, extraer la información ocultada y aplicar análisis estadísticos a la misma (por ejemplo, análisis de entropía para revelar la presencia de información cifrada) [12].

6 Conclusiones. Trabajo Futuro.

El lenguaje natural es un medio poco redundante (ruidoso) en comparación con otros estegomedios más comunes (imágenes, vídeo, etc), esto hace que la ocultación de información de forma imperceptible, estadística y lingüísticamente, no sea tarea sencilla. Por ello, en el presente artículo se analizan las opciones actuales y futuras, y se profundiza en procedimientos de generación automática de estegotextos aplicado a lengua española, en concreto, PCFGs y una propuesta estadística de Peter Wayner.

La utilización de CFGs puede ser una solución de alta calidad (imperceptibilidad lingüística y estadística) en la creación de estegotextos en lengua española. Su problema principal reside en la automatización de la creación de estas gramáticas y en evitar la necesidad (por su criticidad) de que estas tengan que ser privadas y a la vez compartidas entre emisor y receptor de la comunicación enmascarada. Actualmente, la creación de herramientas públicas basadas en estos principios es cuestionable en

términos de seguridad estadística y lingüística. Una futura línea de investigación consiste en innovar en procedimientos públicos que utilizando gramáticas puedan hacer frente a los problemas analizados en el artículo.

Otra propuesta que se analiza en el artículo es un algoritmo de Peter Wayner que genera estegotextos imitando estadísticamente una fuente de texto. La propuesta original permite generar estegotextos en lengua española pero, por las pruebas iniciales, los errores léxicos y sintácticos se producen muy a menudo. Se analiza también una mejora a la idea original (con nivel de atomicidad la palabra) y se implementa en la herramienta libre Stelin (http://stelin.sourceforge.net). Esta variante mejora léxica y sintácticamente la generación de estegotextos en lengua española e incluso produce estegotextos con apariencia semántica. En la práctica aun con esta mejora resulta realmente complicado ocultar más de unas pocas decenas de octetos sin que el estegotexto resultante no tenga problemas, entre los más destacables, repeticiones de expresiones, semánticos o de coherencia global.

La nuevas líneas de investigación abiertas (las que se suponen más prometedoras) consisten en el análisis de las posibilidades sintácticas y semánticas del lenguaje español para la ocultación de información, así como el análisis de procedimientos robustos de sustitución léxica que puedan abrir nuevos caminos, por separado o unidos con las ideas de generación automática de estegotextos, para crear mecanismos robustos que oculten información en textos en lengua española para una información a ocultar de tamaño medio (miles de octetos).

Referencias

1. Carracedo, J.: Seguridad en Redes Telemáticas. Mc-Graw Hill InterAmericana de España. ISBN: 84-481-4157-1 (2004), páginas 123-131.
2. Muñoz, A.: My steganography investigation. http://vototelematico.diatel.upm.es/alfonso.
3. Bergmair, R.: A comprehensive Bibliography of Linguistic Steganography. SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents, *volume 6505, January 2007*.
4. Shamir, A., Van Someren, N: Playing 'Hide and Seek' with Stored Keys. Lecture Notes in Computer Science. Springer Berlin. Volume 1648/1999. ISBN 978-3-540-66362-1.
5. Shirali-Shahreza, M., Shirali-Shahreza, M.H: Text Steganography in SMS. *ICCIT 2007*, Gyeongju, Korea, November 21-23, 2007, pp. 2260-2265.
6. Lingjun, L., Liusheng, H, Xinxin, Zhao., et al: A statistical attack on Kind of Word-Shift Text-Steganography. IIH-MSP 2008. Pages 1503-1507. 2008. ISBN:978-0-7695-3278-3
7. Lingyun, X., Xingming, S., Gang, L., Can, G: Research on Steganalysis fort text steganography based on font format. IAS 2007. Page 490-495. 2007. ISBN: 0-7695-2876-7.
8. Xin-guang, Sui., Hui, Luo., Zhong-liand, Zhu: A steganalysis method based on the distribution of Characters. ICSP 2006. 0-7803-9737-1. 2006 (IEEE)
9. Xin-guang, Sui., Hui, Luo., Zhong-liand, Zhu:A steganalysis method based on the distribution of first letters of words. IIH-MSP 2006. 0-7695-2745-0. IEEE
10. Zhi-li, Chen., Liu-Sheng, Huang., et al: .Effective Linguistic Steganography Detection. IEEE 8th CIT Workshops. 978-0-7695-3242-4. 2008.
11. Blasco, J., Hernandez, J., et al: Csteg: Talking in C code. In Proceedings of SECRYPT International Conference, pag. 399–406. INSTICC. Oporto. July 2008.
12. Hwang, M., A New Redundancy Reducing Cipher. Informatica, vol. 11, no. 4, pp. 435-440, Oct. 2000.