Descripciones lingüísticas de datos de observación meteorológica usando temple simulado

Andrea Cascallar Fuentes, Alejandro Ramos Soto, Alberto J. Bugarín Diz
{andrea.cascallar.fuentes, alejandro.ramos, alberto.bugarin.diz}@usc.es
Centro Singular de Investigación en Tecnoloxías da Información (CiTIUS), Universidade de Santiago de Compostela

***Resumen*—En este trabajo presentamos una aproximación para la generación de descripciones lingüísticas en tiempo real sobre datos de observación meteorológica proporcionados por la Agencia Meteorológica gallega (MeteoGalicia). Las descripciones son sentencias cuantificadas borrosas, que incluyen referencias geográficas imprecisas, y resultan válidas para la etapa de determinación de contenidos de un sistema de Generación de Lenguaje Natural. La generación de las descripciones va guiada por la metaheurística Temple Simulado, que permite seleccionar las descripciones lingüísticas más adecuadas, de acuerdo con un conjunto de criterios objetivos.**

***Index Terms*—Descripciones lingüísticas de datos, Generación de lenguaje natural, Computación con palabras.**

I. Introducción

Ante la ingente cantidad de datos presente actualmente en todas las facetas de la vida, la Inteligencia Artificial provee herramientas que permiten analizar conjuntos de datos con la finalidad de extraer información útil y comprensible para usuarios y expertos, basada en el potencial del lenguaje humano. Por ejemplo, la generación de lenguaje natural (NLG, en inglés) se ocupa de la generación de texto a partir de diversas fuentes de datos [1]. Dentro de este campo son de interés los sistemas *data-to-text* (D2T) [2], que generan textos a partir de conjuntos o series de datos numéricos o simbólicos.

De forma complementaria a los sistemas D2T, las descripciones lingüísticas de datos (LDD, en inglés) proporcionan mecanismos para obtener resúmenes sintéticos de conjuntos de datos numéricos, generalmente basados en el uso de sentencias cuantificadas borrosas [3]. Este tipo de aproximaciones, surgidas a partir del concepto de protoforma definido por Zadeh [4], permiten modelar la imprecisión de los términos lingüísticos inherente al lenguaje humano, aunque su uso en sistemas reales es muy limitada por el momento [5]. Según sus componentes, existen dos tipos de protoformas: tipo I ("Q Y son S") donde Q es un cuantificador, Y es un conjunto de elementos y S es un resumen; y tipo II ("Q KY son S") donde además de los componentes presentes en las de tipo I se añade un calificador K.

Es precisamente en la faceta aplicada de LDD donde se centra el objetivo de este trabajo: la generación de descripciones lingüísticas mediante temple simulado [6] sobre datos meteorológicos en tiempo real proporcionados por MeteoGalicia [7] para el conjunto de los 314 municipios de Galicia, introduciendo además el uso de referencias geográficas.

II. Trabajos relacionados

Actualmente, existen numerosos sistemas NLG, de los cuales una gran mayoría lo componen sistemas D2T [1], [8], [9]. Por ejemplo, uno de los sistemas de mayor impacto en el ámbito de la salud es el sistema BT45, dentro del proyecto BabyTalk [10], que genera informes a partir de datos recogidos durante 45 minutos sobre bebés que se encuentran en la UCI.

En el campo de LDD, la mayor parte de aproximaciones propuestas generan sentencias cuantificadas como "La mayoría de las personas son altas" (tipo I) o "La mayoría de las personas altas son rubias" (tipo II) [3], [9], [11]. Este tipo de propuestas se han aplicado a una gran variedad de casos de uso, principalmente sobre series de datos temporales, como datos de consumo energético [12], fondos de inversión [13], actividad física [14], [15] o el flujo de pacientes en hospitales [16], entre otros. En ámbitos de aplicación real, GALiWeather [5] es un sistema D2T meteorológico que emplea LDD para ciertas tareas de extracción de información, lo que ejemplifica la complementariedad que existe entre ambas disciplinas.

Otro aspecto importante en este campo es la utilización de estrategias de búsqueda, tanto heurísticas [5], [16] como basadas en algoritmos genéticos [17]–[19], para la obtención de sentencias cuantificadas con un nivel de calidad descriptiva suficiente, en problemas donde su número es demasiado grande como para obtenerlas exhaustivamente en su totalidad.

III. Motivación

Nuestro caso de uso consiste en datos de observación meteorológica proporcionados en tiempo cuasi real por MeteoGalicia [7] para las variables estado del cielo, viento y temperatura. Dada la alta frecuencia de actualización de dichos datos, aproximadamente a cada hora, se justifica la necesidad de generar descripciones en el menor tiempo posible. Además, dado que dichos datos se encuentran caracterizados geográficamente, las descripciones en nuestra propuesta contemplan también la inclusión de referencias geográficas vagas como "norte" o "este".

Concretamente, nuestra solución genera descripciones lingüísticas basadas en proposiciones cuantificadas de tipo I, donde Q es un cuantificador, X es una variable lingüística definida a partir de las variables meteorológicas y A es uno de sus valores ("En algunos ayuntamientos el cielo está despejado"); y II donde se añade un descriptor geográfico ("En algunos ayuntamientos en el Norte el cielo está despejado"), que pueden incluir una o más variables meteorológicas.

En nuestro caso, la cantidad de datos disponibles y la necesidad de disponer de una solución computacionalmente poco costosa para la obtención de las descripciones, debido a las restricciones temporales que se deben cumplir, aconsejan la utilización de una estrategia de búsqueda más simple que las de tipo evolutivo comentadas anteriormente. Por ello, proponemos utilizar la metaheurística temple simulado, de muy reducido coste computacional y que ha sido utilizada para abordar diversos problemas [20] [21], obteniendo buenas soluciones en comparación con otras metaheurísticas.

IV. Generación de descripciones meteorológicas

La solución propuesta utiliza datos numéricos para generar descripciones de observación meteorológica en tiempo real de la comunidad autónoma de Galicia.

MeteoGalicia ofrece un servicio web que muestra datos de observación sobre el estado meteorológico actual de los ayuntamientos gallegos [7], actualizados aproximadamente cada hora.

IV-A. Conocimiento del dominio

En base a las variables meteorológicas, se definen las siguientes variables lingüísticas:

- Estado del cielo: sus valores son códigos numéricos que categorizan el la cobertura nubosa y el nivel de precipitación. A partir de esta variable se crea una variable lingüística con el mismo nombre definida del mismo modo.
- Viento: los posibles valores son códigos numéricos que codifican intensidad y dirección del viento. A partir de esta variable se crea una variable lingüística con el mismo nombre que respeta la definición original.
- Temperatura: se crean dos variables lingüísticas, una para las temperaturas máximas y otra para las mínimas. La temperatura actual de un ayuntamiento se compara con las máximas y las mínimas del mes actual del registro de datos históricos. Para la temperatura actual de cada ayuntamiento t_i, las etiquetas que toman las variables lingüísticas se calculan utilizando la media $\bar{x}$ y la desviación típica σ del mes actual de los datos históricos.
  Las protoformas tipo I, siguen la plantilla "En Q ayuntamientos A", donde Q es un cuantificador y A puede ser una o varias de las estructuras definidas en la Tabla I ("En pocos ayuntamientos el estado del cielo es soleado y la temperatura es baja con respecto a las máximas y normal con respecto a las mínimas"). Se han definido siete cuantificadores borrosos ("ninguno", "pocos", "algunos", "aproximadamente la mitad", "bastantes", "casi todos", "todos"), sobre el porcentaje de ayuntamientos (PA) que verifican la sentencia.
  Las protoformas tipo II siguen la plantilla "En Q ayuntamientos en G A", donde Q y A son como en el caso anterior y G es un descriptor geográfico ("En algunos ayuntamientos del Sur el estado del cielo es soleado").

  Los descriptores geográficos, definidos de forma borrosa, son {Norte, Sur, Este, Oeste, Centro} definidos de la siguiente forma:
  - Sur: definido mediante el conjunto borroso trapezoidal de soporte [41.75, 42.57] y núcleo [41.75, 42.16]
  - CentroLatitud: definido mediante el conjunto borroso trapezoidal de soporte [42.16, 43.39] y núcleo [42.57, 42.98]
  - Norte: definido mediante el conjunto borroso trapezoidal de soporte [42.98, 43.8] y núcleo [43.39, 43.8]
  - Oeste: definido mediante el conjunto borroso trapezoidal de soporte [-9.31, -8.266] y núcleo [-9.31, -8.788]
  - CentroLongitud: definido mediante el conjunto borroso trapezoidal de soporte [-8.788, -7.222] y núcleo [-8.266, -7.744]
  - Este: definido mediante el conjunto borroso trapezoidal de soporte [-7.744, -6.7] y núcleo [-7.222, -6.7]

  Para considerar la parte geográfica de la descripción tenemos datos meteorológicos geolocalizados, es decir, se dispone de la localización de cada ayuntamiento. Para cada localización se toma la etiqueta que mejor la representa calculando el grado de cumplimiento aplicando el modelo de cuantificación de Zadeh.

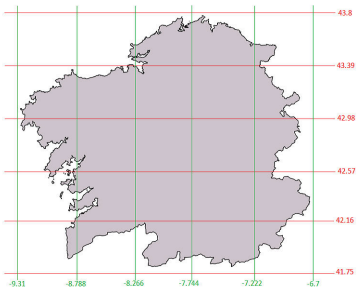


Figura 1: Coordenadas de referencia para la definición de los descriptores geográficos.

Tabla I: Plantillas de texto

Variable lingüística	**Plantilla**
Estado del cielo	el estado del cielo es <valor>
Viento	el viento tiene dirección <valor_dirección> e intensidad <valor_intensidad>
Temperatura	la temperatura es <valor_máx> con respecto a las máximas y <valor_min> con respecto a las mínimas

IV-B. Generación de descripciones lingüísticas

Debido a que el objetivo de nuestra solución es generar sentencias que describan la situación meteorológica en tiempo real, y que la gran mayoría de estaciones meteorológicas sirven datos diezminutales, hemos fijado como restricción que el tiempo de ejecución para la obtención de las descripciones no supere los 5 minutos[1]. Esto se debe a que nuestra solución

[1]Pruebas ejecutadas en un Intel Core i7-6700HQ @2.60GHz 2.59GHz con 16 GB de RAM

se centra en la fase de determinación de contenido de un sistema NLG, por lo tanto, a la hora de abordar las demás fases para desarrollar un sistema completo, el tiempo de ejecución necesario será mayor. Por esta razón y teniendo en cuenta que se busca describir el estado actual en tiempo real, a partir de la versión base hemos introducido una serie de optimizaciones.

En primer lugar, para cada ayuntamiento se calcula el grado de cumplimiento de su temperatura actual con respecto a las etiquetas que pueden tomar las dos variables lingüísticas. Para realizar este cálculo, utilizando el valor de la media y la desviación típica, para cada ayuntamiento se guarda el grado de cumplimiento para cada una de las etiquetas definidas en la Sección III. Además, para cada posible valor de la variable "temperatura" L, se calcula el grado de pertenencia del conjunto de ayuntamientos como se muestra en la expresión 1 siendo n el número de ayuntamientos, μ_L la función que evalúa el grado de cumplimiento y t_i la temperatura actual de cada ayuntamiento.

$$\mu(L) = \frac{\sum_{i=1}^{|n|} \mu_L(t_i)}{n} \tag{1}$$

IV-B1. Generación de sentencias tipo I: Esta parte del sistema se centra en generar las sentencias de tipo I a partir de los datos obtenidos.

Versión inicial. Genera todas las sentencias posibles y, para cada una de ellas, se calcula su grado de cumplimiento. Este cálculo se realiza de forma diferente si la sentencia contiene una única variable lingüística o si está compuesta por más de una.

Para las descripciones D donde solo se describe una variable se aplica el modelo de cuantificación de Zadeh. En el caso de las variables "crisp" el grado de cumplimiento es 1 si el valor actual v_i coincide con la etiqueta S y 0 en caso contrario.

$$\mu(Q\ X\ son\ A) = Q(\frac{\sum_{i=1}^{|n|} \mu_S(v_i)}{n}) \tag{2}$$

Para las descripciones D donde se describe más de una variable, debemos aplicar, siguiendo nuevamente el modelo de cuantificación de Zadeh, la conjunción de todas las variables, utilizando la t-norma mínimo:

$$\mu(Q\ X\ son\ A) = Q(\frac{\sum_{i=1}^{|n|} \mu_{S1}(v_{S1i}) \cap ... \cap \mu_{Sm}(v_{Smi})}{n}) \tag{3}$$

Al ejecutar esta versión se obtiene un total de 63.875 descripciones y la ejecución tarda en completarse aproximadamente 35 minutos.

Optimización 1. La versión anterior no cumple la restricción, por lo tanto es necesario optimizarla.

Por muy diversas que sean las condiciones meteorológicas, no se van a dar todos los posibles valores para las variables contempladas. Esto es debido, en parte, al tamaño reducido de esta región.

Para reducir el coste temporal que supone generar todas las posibles combinaciones, se propone implementar una optimización que descarte los valores no presentes en el panorama actual. De este modo se reduce notoriamente el tiempo necesario para completar una ejecución. El tiempo necesario para obtener las descripciones varía entre 50 y 60 segundos y se obtienen alrededor de 2000. Además, eliminando situaciones meteorológicas que no están presentes se obtienen descripciones más representativas del mapa.

Mejores soluciones. Aplicando esta optimización se consigue una mejora notoria, sin embargo, los resultados obtenidos no son suficientemente representativos. Para solucionar este problema se proponen las siguientes mejoras:

- Eliminar descripciones "Ninguno": este cuantificador es útil en cuanto que aísla las descripciones que no se cumplen, sin embargo, éstas no son, en general, útiles para el usuario, por lo tanto, se descartan.
- Umbral en el grado de cumplimiento: algunas sentencias describen casos poco relevantes, esto es, tienen un grado de cumplimiento muy bajo. Para evitar esta situación se define un umbral $u = 0{,}5$ eliminando aquellas sentencias que tengan un grado de cumplimiento inferior.
- Ordenación: para evaluar las descripciones se propone utilizar dos criterios además del grado de cumplimiento: cobertura del cuantificador, prefiriendo sentencias que abarquen mayor extensión y así evitar describir una misma situación mediante varias sentencias referidas a extensiones más reducidas; y tamaño de la sentencia, prefiriendo sentencias que comprendan un número mayor de variables, ya que de esta forma se reduce el número de sentencias y estas son más específicas. Las sentencias se ordenan según los criterios descritos anteriormente, priorizados por el siguiente orden: grado de cumplimiento, cobertura del cuantificador y longitud de la sentencia. Una excepción en dicho orden son las descripciones del cuantificador "Pocos", que se sitúan después de las de cuantificadores con mayor cobertura de modo que, aunque según la ordenación indicada podrían ser seleccionadas en detrimento de algunas sentencias con mayor cobertura, se colocan después de modo que el usuario obtenga información más general y pueda acceder a éstas si necesita más grado de detalle.

Una vez que se han aplicado estas mejoras, se define un número máximo de descripciones que se van a mostrar ya que un número elevado de descripciones puede provocar que el texto generado no sea útil para el usuario.

En la Tabla II se muestra una comparativa entre las diferentes versiones.

Tabla II: Resumen versiones tipo I

Versión	Tamaño solución	Tiempo de ejecución
Inicial	63875	∼35 minutos
Optimización 1	∼2000	50-60 segundos
Mejores soluciones	un máximo de 100	50-60 segundos

IV-B2. Generación de sentencias tipo II: Se parte de una versión inicial, donde se generan todas las combinaciones posibles, y a partir de ahí se proponen optimizaciones.

Para esta descripciones hay que calcular, para cada ayuntamiento, los grados de pertenencia con respecto a los descriptores geográficos. Este cálculo se realiza utilizando la coordenada correspondiente y calculado su grado de cumplimiento. Los cálculos para "Centro" difieren del resto ya que está formado por dos componentes. El proceso es el siguiente: para cada ayuntamiento se calcula el grado de cumplimiento para cada coordenada, se aplica la t-norma mínimo entre estos dos valores y al valor resultante se le aplica el descriptor geográfico, calculando el grado de cumplimiento.

$$\mu_{Centro(longitud_i,latitud_i)} = \mu_{CentroLongitud(longitud_i)} \cap \mu_{CentroLatitud(latitud_i)} \quad (4)$$

Versión inicial. Genera todas las sentencias posibles y, para cada una de ellas, se calcula su grado de cumplimiento con la finalidad de saber cuáles son las descripciones más representativas.

$$\mu(Q\ X\ son\ A\ en\ G) = Q(\frac{\sum_{i=1}^{|n|} \mu_G(x_i) \cap \mu_{S1}(x_i) \cap ... \cap \mu_{Sm}(x_i)}{\sum_{i=1}^{|n|}(\mu_G(x_i))}) \quad (5)$$

Para obtener todas las descripciones, el sistema necesita aproximadamente 3 días y se generan 1094170 descripciones, incumpliendo la restricción temporal.

Optimización 1. La primera optimización es eliminar los valores no presentes, como en las de tipo I, reduciendo el coste temporal a aproximadamente 8 horas y obteniendo alrededor de 13000 descripciones.

V. Algoritmo de búsqueda meta-heurístico

Debido a la elevada cantidad de descripciones que se generan de tipo II y al coste temporal que esto supone, se propone el uso de un algoritmo de búsqueda metaheurística para la extracción de información relevante consumiendo menos recursos.

En general, los algoritmos basados en poblaciones no parecen la mejor opción para este caso. Los algoritmos metaheurísticos basados en métodos constructivos tampoco son una buena opción ya que, en este caso, la solución inicial debe tener un mínimo de componentes. Las soluciones basadas en trayectorias pueden aplicarse en este caso, ya que utilizan una heurística de búsqueda local que explora posibles soluciones siguiendo una trayectoria en el espacio de búsqueda. Realizando un análisis de los algoritmos más utilizados se concluye que el Temple Simulado [6] es una buena alternativa. Es un algoritmo de búsqueda por entornos con un criterio probabilístico de aceptación de soluciones inspirado en la Termodinámica. En cada iteración se genera un determinado número de vecinos, con cierta probabilidad de aceptar soluciones peores para evitar que el algoritmo no se estanca en un óptimo local.

Este algoritmo cuenta con una serie de parámetros configurables de modo que, realizando un estudio empírico, se puedan establecer valores que obtengan resultados de calidad. A continuación se muestran los valores establecidos después de realizar el estudio.

- Valor inicial del parámetro de control T_0: esta variable se debe inicializar a un valor suficientemente alto ya que, si es muy bajo converge demasiado rápido y si es muy alto tarda en converger. Después de experimentar con diversos valores se inicializa con un valor proporcional al número máximo de descripciones posible, tomando como parámetros $\mu = 0{,}01$ y $\phi = 0{,}999$.

$$T_0 = (\mu / -\ln(\phi))/MAX\_SOLUCIONES \quad (6)$$

- Solución inicial S_0: después de probar diversas alternativas, S_0 se inicializa con la mejor sentencia formada por una única variable lingüística obtenida aplicando los criterios de evaluación descritos anteriormente.
- Nueva solución: la forma de generar una nueva solución es modificar [1, 4] componentes aleatoriamente. Con este método en ocasiones el algoritmo se queda estancado, para solucionar esto, se define un parámetro $maxRepeated$, que define el número máximo de soluciones repetidas en una iteración. Si se alcanza este valor, se genera una nueva solución completamente aleatoria, evitando el estancamiento.
- Velocidad y mecanismo de enfriamiento: $MAX\_CANDIDATAS = 300$ y $MAX\_ACEPTADAS = 30$ se consigue un buen balance en cuanto a soluciones peores aceptadas y la velocidad de enfriamiento. En cuanto al mecanismo de enfriamiento, se utiliza el Esquema de Cauchy.
- Condición de parada: después de experimentar con otras alternativas que no se adecúan a este problema, se experimenta con una condición de parada que está formada por dos condiciones: número máximo de iteraciones y máximo de intentos fallidos de generar nuevas soluciones para cada iteración. Para esto se definen dos parámetros: $MAX\_ITER = 2300$, que establece el número máximo de iteraciones que el algoritmo puede hacer y $MAX\_NEIGHBOUR\_ATTEMPTS = 2000$, que define el número máximo de intentos de generar una nueva solución candidata en una iteración.
- Condición de aceptación: realizamos una experimentación con diferentes opciones concluyendo que la que mejor resultados ofrece es la expresión 7, donde se da preferencia al grado de cumplimiento, aceptando las nuevas soluciones que mejoren el de la actual, y a la cobertura, aceptando soluciones con mayor cobertura aunque el grado de cumplimiento sea inferior.

$$\begin{aligned}(cobertura(S_{cand}) <= cobertura(S_{act})\ \&\&\\ \mu(S_{cand}) >= \mu(S_{act})\ \&\&\ \mu(S_{cand} > 0))\ ||\\ (cobertura(S_{cand}) > cobertura(S_{act})\ \&\&\\ \mu(S_{cand} > 0))\ ||\ (\mu(S_{cand} > 0))\ \&\&\\ aleatorio < e^{-\delta/T_k})\end{aligned} \quad (7)$$

A esta configuración se le aplican las mismas optimizaciones que en las descripciones de tipo I descritas en la Sección IV-B1:

Aplicando este algoritmo, el tiempo de ejecución es de aproximadamente 2 minutos y el conjunto de descripciones que se muestra tiene un tamaño de aproximadamente 400 sentencias ofreciendo una descripción representativa del estado meteorológico.

Tabla III: Resumen versiones tipo II

Versión	Tamaño solución	Tiempo de ejecución
Inicial	1094170	3 días
Optimización 1	∼13000	∼8 horas
SA	∼400	∼2 minutos

VI. Evaluación

Para evaluar el grado de adecuación de las descripciones a diferentes casos hemos generado cuatro mapas, en cada uno de los cuales se representa el estado de una de las variables (estado del cielo, viento, temperatura máxima y temperatura mínima). Se representa un meteoro por cada uno de los 314 ayuntamientos de Galicia.

Para las variables "estado del cielo" y "viento" los mapas se construyen utilizando los iconos que ofrece MeteoGalicia. Para los componentes de la variable "temperatura" se utilizan los siguientes códigos que representan las posibles etiquetas de la variable lingüística "temperatura": "MB" para "muy baja", "B" para "baja", "N" para "normal", "A" para "alta" y "MA" para "muy alta". Cada uno de estos códigos tiene un color desde azul oscuro para "MB" hasta rojo para "MA". Debido al elevado número de iconos, interpretar correctamente la información visual es una tarea difícil.

En las tablas IV y V se muestran las mejores descripciones con respecto a los mapas que se muestran en la Figura 2. Estas descripciones son las 10 mejores obtenidas en cada caso después de ordenarlas siguiendo los tres criterios, descritos en IV-B1. Aunque las descripciones de tipo I proporcionan una idea general sobre el estado meteorológico, por sí solas no son suficientemente informativas. En este sentido, las descripciones de tipo II son de mayor utilidad ya que, al centrarse en regiones más pequeñas tienen en cuenta situaciones que las de tipo I no consideran precisamente por su carácter general.

Para comprobar la calidad de las descripciones se han realizado 20 ejecuciones en momentos del día diferentes durante dos semanas. Las descripciones de tipo I siempre obtuvieron buenos resultados en cuanto a que describen la situación meteorológica informando sobre los valores de las variables meteorológicas que más se repiten. Las descripciones de tipo II, en general también obtuvieron resultados representativos salvo en casos en los que un fenómeno ocurre en un área muy pequeña debido a la definición demasiado amplia del descriptor geográfico (por ejemplo cielos soleados en toda la región salvo en el extremo Norte de Galicia donde está "muy nublado") de modo que no ocupa una posición relevante en el conjunto de soluciones resultante.

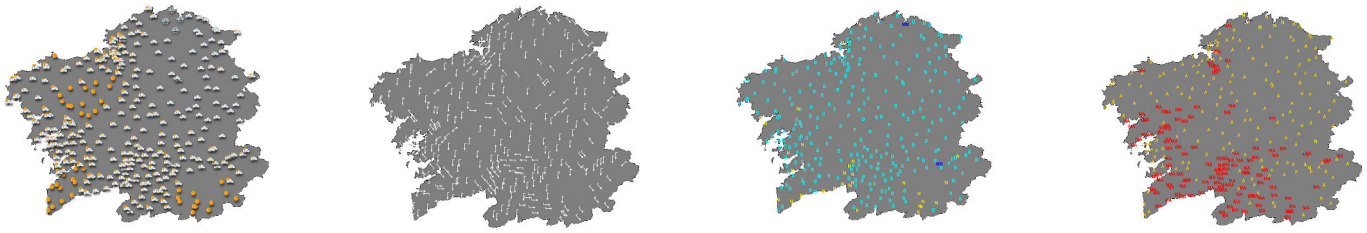
Tabla IV: Mejores 10 descripciones tipo I para 15 de septiembre a las 17:00 (Figura 2).

Descripción
En aproximadamente la mitad de ayuntamientos el estado del cielo es muy nublado
En aproximadamente la mitad de ayuntamientos el viento tiene dirección Norte e intensidad baja
En algunos ayuntamientos el cielo está muy nublado y la temperatura es baja con respecto a las máximas y alta con respecto a las mínimas
En algunos ayuntamientos el viento tiene dirección Norte e intensidad baja y la temperatura es baja con respecto a las máximas y alta con respecto a las mínimas
En algunos ayuntamientos la temperatura es baja con respecto a las máximas y muy alta con respecto a las mínimas
En bastantes ayuntamientos la temperatura es baja con respecto a las máximas y alta con respecto a las mínimas
En algunos ayuntamientos el estado del cielo es nubes y claros
En algunos ayuntamientos el estado del cielo es muy nublado y el viento tiene dirección Norte e intensidad baja
En pocos ayuntamientos el estado del cielo es despejado y el viento tiene dirección Norte e intensidad baja

Tabla V: Mejores 10 descripciones tipo II para 15 de septiembre a las 17:00 (Figura 2).

Descipción
En aproximadamente la mitad de ayuntamientos del Este la temperatura es baja con respecto a las máximas y alta con respecto a las mínimas
En algunos ayuntamientos del Oeste el viento tiene dirección Norte e intensidad baja
En algunos ayuntamientos del Oeste el estado del cielo es nubes y claros
En algunos ayuntamientos del Oeste el estado del cielo es muy nublado
En algunos ayuntamientos del Sur el viento tiene dirección Norte e intensidad baja
En algunos ayuntamientos del Sur la temperatura es baja con respecto a las máximas y alta con respecto a las mínimas
En aproximadamente la mitad de ayuntamientos del Norte la temperatura es baja con respecto a las máximas y alta con respecto a las mínimas
En algunos ayuntamientos del Este el viento tiene dirección Norte e intensidad baja
En algunos ayuntamientos del Este el estado del cielo es cubierto y la temperatura es baja con respecto a las máximas y alta con respecto a las mínimas
En algunos ayuntamientos del Sur el estado del cielo es muy nublado y la temperatura es baja con respecto a las máximas y muy alta con respecto a las mínimas

Hemos analizado cuantitativamente la calidad de las descripciones generadas con la metaheurística frente a la totalidad de descripciones posibles (en casos de uso donde esto último resultaba factible). Los resultados indican que la metaheurística genera: i) el 25 % de las descripciones de mayor calidad (utilizando como métrica de calidad la suma de las tres condiciones de evaluación normalizadas) si descartamos las que incluyen el cuantificador de menor cobertura espacial (el menos específico, "pocos ayuntamientos"), ii) el 40 % de las descripciones que incluyen los tres cuantificadores de mayor cobertura y iii) el 75 % de las descripciones que incluyen los dos cuantificadores de mayor cobertura ("casi todos" y "bastantes"). Por lo tanto, se puede concluir que la metaheurística es efectiva para los cuantificadores que suponen mayor cobertura espacial (los más específicos).



(a) Estado del cielo (b) Viento (c) Temperaturas máximas (d) Temperaturas mínimas

Figura 2: Mapas que definen el estado meteorológico del 15 de septiembre de 2017 a las 17:00 descrito en IV y V.

Por último, hemos evaluado la eficacia de la metaheurística, comparando la calidad de la solución final frente a la solución inicial. Aquí se observa que la media de mejora de las soluciones es del 25 % para cinco ejecuciones realizadas.

VII. Conclusiones

En este trabajo hemos presentado una aproximación basada en fuerza bruta para la generación de descripciones lingüísticas compuestas de sentencias cuantificadas de tipo I y temple simulado para las de tipo II. Dichas descripciones se generan a partir de datos meteorológicos de observación, sobre un conjunto de variables lingüísticas tanto *crisp* como borrosas, entre las que destacan la inclusión de referencias geográficas. Hemos comparado las sentencias obtenidas con mapas para comprobar si eran representativas. Además, para el caso de las sentencias tipo II, se ha evaluado que la metaheurística genera entre el 25 % y el 75 % de las descripciones de mayor calidad de entre todas las posibles. La metaheurística también resulta efectiva, ya que la calidad de la solución final frente a la inicial mejora en un 25 % en promedio.

Como trabajo futuro, ampliaremos el modelo para nuevas metaheurísticas que puedan compararse en cuanto a rendimiento y calidad de las descripciones generadas.

Agradecimientos

Este trabajo ha sido financiado por el Ministerio de Economía y Competitividad (proyectos TIN2014-56633-C3-1-R y TIN2017-84796-C2-1-R) y la Consellería de Educación de la Xunta de Galicia (proyectos GRC2014/030 y Acreditación 2016-2019, ED431G/08"). Todos los proyectos fueron cofinanciados por el programa FEDER. A. Ramos-Soto agradece la financiación de la Consellería de Cultura, Educación e Ordenación Universitaria" (Beca Postdoctoral ED481B 2017/030).

Referencias

[1] A. Gatt and E. Krahmer, "Survey of the state of the art in natural language generation: Core tasks, applications and evaluation," *Journal of Artificial Intelligence Research*, vol. 61, pp. 65–170, 2018.

[2] E. Reiter, "An architecture for data-to-text systems," in *Proceedings of the Eleventh European Workshop on Natural Language Generation*. Association for Computational Linguistics, 2007, pp. 97–104.

[3] N. Marín and D. Sánchez, "On generating linguistic descriptions of time series," *Fuzzy Sets and Systems*, vol. 285, pp. 6–30, 2016.

[4] L. A. Zadeh, "A prototype-centered approach to adding deduction capability to search engines-the concept of protoform," in *Intelligent Systems, 2002. Proceedings. 2002 First International IEEE Symposium*, vol. 1. IEEE, 2002, pp. 2–3.

[5] A. Ramos-Soto, A. J. Bugarin, S. Barro, and J. Taboada, "Linguistic descriptions for automatic generation of textual short-term weather forecasts on real prediction data," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 1, pp. 44–57, 2015.

[6] P. J. Van Laarhoven and E. H. Aarts, "Simulated annealing," in *Simulated annealing: Theory and applications*. Springer, 1987, pp. 7–15.

[7] MeteoGalicia: servicio de datos de observación en tiempo real. (2017). [Online]. Available: http://servizos.meteogalicia.gal/rss/observacion/observacionConcellos.action

[8] J. Bateman and M.Zock. NLG systems wiki. (2017). [Online]. Available: http://nlg-wiki.org/systems/

[9] A. Ramos-Soto, A. Bugarín, and S. Barro, "On the role of linguistic descriptions of data in the building of natural language generation systems," *Fuzzy Sets and Systems*, vol. 285, pp. 31–51, 2016.

[10] F. Portet, E. Reiter, A. Gatt, J. Hunter, S. Sripada, Y. Freer, and C. Sykes, "Automatic generation of textual summaries from neonatal intensive care data," *Artificial Intelligence*, vol. 173, no. 7-8, pp. 789–816, 2009.

[11] J. Kacprzyk and S. Zadrozny, "Computing with words is an implementable paradigm: fuzzy queries, linguistic data summaries, and natural-language generation," *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 3, pp. 461–472, 2010.

[12] A. van der Heide and G. Triviño, "Automatically generated linguistic summaries of energy consumption data," in *Intelligent Systems Design and Applications, 2009. ISDA'09. Ninth International Conference on*. IEEE, 2009, pp. 553–559.

[13] J. Kacprzyk and A. Wilbik, "Using fuzzy linguistic summaries for the comparison of time series: an application to the analysis of investment fund quotations." in *IFSA/EUSFLAT Conf.*, 2009, pp. 1321–1326.

[14] D. Sanchez-Valdes, L. Eciolaza, and G. Trivino, "Linguistic description of human activity based on mobile phone's accelerometers." in *IWAAL*. Springer, 2012, pp. 346–353.

[15] A. Alvarez-Alvarez and G. Trivino, "Linguistic description of the human gait quality," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 1, pp. 13–23, 2013.

[16] R. M. Castillo-Ortega, N. Marín, and D. Sánchez, "A Fuzzy Approach to the Linguistic Summarization of Time Series." *Journal of Multiple-Valued Logic & Soft Computing*, vol. 17, 2011.

[17] R. Castillo-Ortega, N. Marín, D. Sánchez, and A. G. Tettamanzi, "Linguistic summarization of time series data using genetic algorithms," in *EUSFLAT*, vol. 1, no. 1. Atlantis Press, 2011, pp. 416–423.

[18] C. A. Donis-Díaz, R. Bello, and J. Kacprzyk, "Using ant colony optimization and genetic algorithms for the linguistic summarization of creep data," in *Intelligent Systems' 2014*. Springer, 2015, pp. 81–92.

[19] T. Altintop, R. R. Yager, D. Akay, F. E. Boran, and M. Ünal, "Fuzzy linguistic summarization with genetic algorithm: An application with operational and financial healthcare data," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 25, no. 04, pp. 599–620, 2017.

[20] R. Tavakkoli-Moghaddam, M.-B. Aryanezhad, N. Safaei, and A. Azaron, "Solving a dynamic cell formation problem using metaheuristics," *Applied Mathematics and Computation*, vol. 170, no. 2, pp. 761–780, 2005.

[21] S.-W. Lin, J. Gupta, K.-C. Ying, and Z.-J. Lee, "Using simulated annealing to schedule a flowshop manufacturing cell with sequence-dependent family setup times," *International Journal of Production Research*, vol. 47, pp. 3205–3217, june 2009.