Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación

IIC2613 – Inteligencia Artificial (I/2018)

Tarea 2: análisis de sentimientos usando aprendizaje supervisado

Introducción: como ser racista sin quererlo

El análisis y clasificación de sentimientos es una de las tareas más comunes en el área de procesamiento de lenguaje natural (NLP). Consiste en tomar una palabra o frase, y predecir si el sentimiento asociado a ella es positivo o negativo. Por ejemplo, al evaluar el sentimiento de publicaciones en redes sociales, se puede estimar de manera rápida la reacción de las personas frente a algún evento en particular.

En esta tarea, deberán aplicar todo el conocimiento que han adquirido en aprendizaje supervisado, para construir un sistema que permita obtener la fuerza (negativa o positiva) del sentimiento asociado a palabras o frases en inglés. Luego, una vez construido el sistema, evaluarán su potencial de aplicación mediante la cuantificación de su nivel de prejuicio frente a temas como género y raza. Finalmente, utilizando los (sorprendentes, impactantes y después de pensarlo, razonables pero condenables) resultados, deberán implementar algún mecanismo para disminuir el nivel de prejuicio detectado.

Representación del texto

Existen múltiples manera de representar texto de manera vectorial, con el fin de ser procesado por algoritmos de aprendizaje de máquina. Para esta tarea utilizaremos una representación conocida como *word-embedding*, donde cada palabra es codificada como un vector n-dimensional denso, que vive en un espacio que captura similitudes semánticas entre palabras (más información acá).

Para utilizar un *embedding*, se requiere tener una *matriz de embedding*, donde cada fila representa el vector de una palabra dada. Es importante notar que esto define de manera implícita un vocabulario, por lo que aquellas palabras que no estén en el, no tendrán una representación en el *embedding*.

En particular, para esta tarea, se puede utilizar cualquier embedding disponible en la red, pero recomienda utilizar **GloVe**. Existen tres versiones de este *embedding* disponibles para descargar en `https://nlp.stanford.edu/projects/glove/`. Se recomienda utilizar la versión *Common Crawl 42B*. Independiente de la versión que elijan, descargarán un archivo de texto codificado en *utf-8*, a partir del cual deberán construir la *matriz de embedding*. Dado que esta matriz debe ser indexada usando palabras, se recomienda modelarla utilizando un `DataFrame` de Pandas, o un diccionario.

Set de datos

La fuente primaria de datos para entrenar los modelos será el *opinion lexicon* disponible en `https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon`. Este contiene aproximadamente 6800 palabras, rotuladas como positivas o negativas. Si lo desea, puede utilizar otros conjuntos de datos, siempre y cuando tengan rotulado el sentimiento asociado a la palabra.

Una vez elegido el set de datos, debe codificar cada una de las palabras utilizando la *matriz de embedding*. Recuerde que es altamente probable que existan palabras en el set de datos que no forman parte del vocabulario del embedding. Queda a criterio de ud. el que hacer con estas palabras.

Entrenamiento de los modelos

Una vez codificados los datos, se deben entrenar al menos tres algoritmos de clasificación con ellos. Puede utilizar cualquier algoritmo que crea conveniente, no estando limitados estos a los vistos en clases. Con el fin de facilitar su labor, se recomienda utiliza la biblioteca *scikit-learn*, que provee implementaciones eficientes de una gran cantidad de algoritmos de clasificación.

Es sumamente importante que los modelos obtenidos en esta parte de la tarea cumplan con las siguientes características:

- **Alto rendimiento**: el puntaje de clasificación promedio debe ser sustancialmente superior a la clasificación realizada por una decisión aleatoria.
- **Alto poder de generalización**: el puntaje de clasificación promedio en datos no vistos no debe ser sustancialmente diferente del obtenido en los datos utilizados para entrenar.
- **Salida continua**: cada uno de los modelos generados, debe ser capaz de entregar un puntaje que indique la fortaleza del sentimiento detectado. Para esto pueden utilizarse representaciones probabilísticas o puntajes de clasificación. Es posible que esto requiera el post-procesamiento de la salida del algoritmo usado para la clasificación.

Representación de frases y evaluación de su sentimiento

Con el fin de analizar sentimientos a un nivel de mayor abstracción, debe adaptar su representación y/o sus sistemas de clasificación, para poder procesar no sólo palabras, sino que también frase de largo arbitrario.

Al igual que para el caso de las palabras, los algoritmos deben ser capaces de entregar una salida continua, que cuantifique la fuerza del sentimiento detectado en la frase.

Evaluación y caracterización de prejuicios

Una vez construidos los modelos, evalúe el nivel de prejuicio de ellos, utilizando palabras o frases que involucren temas que son propensos a caer en prejuicios. En particular, se pide explorar, al menos, el nivel de racismo del sistema. Para esto, se recomienda utilizar afirmaciones neutras (frases o palabras), cuyas diferencias radiquen sólo en nombres, lugares o nacionalidades. Por ejemplo, dada la frase "Let's go get * food", se recomienda explorar las variaciones en el puntaje de sentimiento, al sustituir el * por distintos tipos de cocina ("Italian", "Chinese", "Mexican", etc). En el caso de nombres, una estrategia válida es evaluar el sentimiento de nombres asociados a distintas procedencias étnicas (blanco, negro, latino, etc).

Para cada modelo, cuantifique su nivel de prejuicio mediante la definición de un *puntaje de prejuicio* y luego implemente algún mecanismo para disminuir este puntaje, ya sea modificando los modelos y/o los datos (esto puede incluir el *embedding*). Puede complementar y dar más fuerza a su análisis utilizando gráficos y tablas.

Entrega

La tarea debe desarrollarse en Jupyter Notebook con Python 3.x. En el notebook debe ir tanto el código como un informe (preferiblemente intercalados), donde se expliquen los pasos realizados, se analicen los resultados y se planteen conclusiones. La entrega de la tarea tiene como fecha límite el viernes 22 de junio a las 23:59, y debe realizarse en el repositorio en GitHub asignado a cada uno. Para fines de corrección, se revisará la última versión subida al repositorio. En caso de atraso, se aplicará un descuento de 1.0 ptos. cada 6 horas o fracción.

Política de Integridad Académica

Los alumnos de la Escuela de Ingeniería deben mantener un comportamiento acorde al Código de Honor de la Universidad:

"Como miembro de la comunidad de la Pontificia Universidad Católica de Chile me comprometo a respetar los principios y normativas que la rigen. Asimismo, prometo actuar con rectitud y honestidad en las relaciones con los demás integrantes de la comunidad y en la realización de todo trabajo, particularmente en aquellas actividades vinculadas a la docencia, el aprendizaje y la creación, difusión y transferencia del conocimiento. Además, velaré por la integridad de las personas y cuidaré los bienes de la Universidad."

En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los alumnos que incurran en este tipo de acciones se exponen a un procedimiento sumario. Ejemplos de actos deshonestos son la copia, el uso de material o equipos no permitidos en las evaluaciones, el plagio, o la falsificación de identidad, entre otros. Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica en relación a copia y plagio: Todo trabajo presentado por un alumno (grupo) para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno (grupo), sin apoyo en material de terceros. Si un alumno (grupo) copia un trabajo, se le calificará con nota 1.0 en dicha evaluación y dependiendo de la gravedad de sus acciones podrá tener un 1.0 en todo ese ítem de evaluaciones o un 1.1 en el curso. Además, los antecedentes serán enviados a la Dirección de Docencia de la Escuela de Ingeniería para evaluar posteriores sanciones en conjunto con la Universidad, las que pueden incluir un procedimiento sumario. Por "copia" o "plagio" se entiende incluir en el trabajo presentado como propio, partes desarrolladas por otra persona. Está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la cita correspondiente.