TASS 2017: Workshop on Semantic Analysis at SEPLN, septiembre 2017, págs. 65-69

FastText como alternativa a la utilización de Deep Learning en corpus pequeños

FastText as an alternative to using Deep Learning in small corpus

Rosa María Montañés Salas, Rafael del-Hoyo Alonso, Jorge Vea-Murguía Merck, Rocío Aznar Gimeno, Francisco José Lacueva-Pérez
ITAINNOVA
C/ María de Luna, nº 7. 50018 Zaragoza, Spain
{rmontanes, rdelhoyo, jveamurguia, raznar, fjlacueva}@itainnova.es

Resumen: La utilización de deep learning para el análisis del lenguaje natural está siendo una auténtica revolución en este campo. Aunque su calidad está fuera de toda duda, existen un conjunto de técnicas alternativas, con una gran validez cuando el tamaño del corpus de base es pequeño, y donde el uso de técnicas de deep learning no son suficientemente adecuadas, bien por la falta de un conjunto de entrenamiento suficientemente amplio o por la escasez de elevadas capacidades de computación. Fasttext, desarrollada por Joulin et al.(2017), es una solución alternativa al aprendizaje profundo, con resultados comparables, basado en las técnicas de BoW.
Palabras clave: BoW, Deep Learning, aprendizaje automático, análisis de opinión

Abstract: The use of deep learning in natural language processing is being a real revolution in this field. Although its quality is beyond doubt, there is a set of alternative techniques, with great validity when the size of the corpus is small, and where the use of deep learning techniques are not adequate enough due to the lack of big data sets or limited computation power availability. Fasttext, developed by Joulin et al. (2017), is an alternative solution to deep learning, with comparable results, based on BoW techniques.
Keywords: BoW, bag of words, affective, FastText, Deep Learning, Machine Learning, opinion analysis

1 Introducción

1.1 Descripción del Problema

La competición TASS (Martínez-Cámara et al, 2015, 2017) cada año se convierte en el lugar perfecto para la experimentación de nuevas técnicas, y un foro donde compartir experiencias sobre nuevas metodologías para el análisis de algoritmos para la detección de opinión. En esta edición de 2017 dentro de nuestro grupo, hemos decidido afrontar una problemática real que nos ocurre cuando se genera un nuevo modelo de opinión para un sector específico. El corpus de entrenamiento es normalmente de un tamaño reducido, debido a que es costoso generar un corpus de elevado tamaño, pero también es necesario obtener rápidamente resultados. Por una parte, las técnicas de Deep Learning están barriendo en modelos de análisis lingüístico, a través de técnicas como las *Convolutional Neural Networks* (CNN) (Poria et al., 2016), las redes recurrentes Long short-term memory (LSTM) (Liu et al., 2015), Word2Vec (Poria et al., 2016), la captura de los efectos del árbol sintáctico en el modelo *Recursive Neural Tensor Network* (RNTN), (Socher et al., 2013) o un nuevo modelo como Deepmind presentado en 2017 (Radford et al., 2017). Éste último fue probado para esta edición pero los resultados obtenidos fueron muy pobres. La mayoría de estos modelos están encontrando grandes resultados con grandes corpus, lo que hace que, por otro lado, los modelos de bolsas de palabras (BoW), el ensamblado de algoritmos (del Hoyo et al., 2015) y la extracción de variables específicas tengan todavía cabida como una de las soluciones válidas en competiciones como el TASS y para el desarrollo de proyectos con corpus específicos y de menor tamaño.

ISSN 1613-0073

En este artículo se presenta la arquitectura utilizada, el algoritmo FastText y cómo se ha combinado con el uso de herramientas propias para resolver la primera tarea propuesta en el TASS 2017: "Análisis de Sentimiento a nivel de tweet". A continuación se presentan los resultados obtenidos y finalmente se generan unas conclusiones generales.

2 Arquitectura

2.1 Descripción del algoritmo FastText

FastText es una librería desarrollada sobre C++ para el aprendizaje eficiente de representaciones de palabras y clasificación de textos. Los experimentos realizados por parte del grupo de Facebook AI Research (Joulin et al., 2017) demuestran que el clasificador FastText ofrece resultados equiparables a los obtenidos mediante clasificadores de deep learning en términos de precisión, siendo éste mucho más rápido sin necesidad de ser ejecutado sobre GPU.

La arquitectura de FastText se basa en el modelo de bolsas de palabras (BoW) mejorado a través del uso de clasificadores basados en redes neuronales multicapa (MLNN), en lugar de los tradicionales clasificadores lineales cuya capacidad de generalización suele ser bastante pobre. El uso de MLNN permite compartir información entre las variables y las clases a través de la capa oculta. La arquitectura implementada se asemeja a la desarrollada por (Mikolov et al., 2013) para la representación de palabras en un espacio vectorial n-dimensional.

En concreto, el modelo está basado en el algoritmo CBoW (Continuous BoW), red neuronal de una capa oculta, e incluye información de Ngramas (Bag of n-gramas) consiguiendo así capturar información respecto al orden de las palabras. Para conseguir una mayor eficiencia computacional se utiliza una capa softmax jerárquica en la capa de salida. Este algoritmo se ha utilizado en el caso de estudio para entrenar un clasificador para análisis de sentimiento sobre tweets en castellano previamente procesados. Los modelos generados han sido validados con el conjunto de test proporcionado (corpus de desarrollo de TASS) obteniendo la precisión y el recall mediante las funcionalidades expuestas por la librería. Finalmente, los mejores modelos han sido evaluados con el corpus de test provisto por el TASS este año (InterTASS).

2.2 Reducción de la dimensión del problema

La generación de modelos de lenguaje requiere de corpus de gran tamaño para el entrenamiento de algoritmos y obtención de modelos precisos capaces de capturar las sutilezas del lenguaje hablado y escrito. En muchos casos, no se dispone de dicha cantidad de información y es necesario recurrir a conjuntos de datos más reducidos y enfocados a campos específicos, aún en esta situación la información textual de los documentos puede ser redundante y contener términos de baja utilidad en cuanto a lo que al aprendizaje se refiere. Por ello se plantea el uso de técnicas de procesado de texto para realizar una limpieza de los datos y disminuir la dimensionalidad del problema.

Moriarty® (Peña et al., 2016) es la herramienta de diseño e implementación de soluciones avanzadas de software de Inteligencia Artificial, desarrollada por ITAINNOVA. A través de su interfaz web se pueden diseñar y poner en marcha flujos de trabajo para el tratamiento de todo tipo de datos.

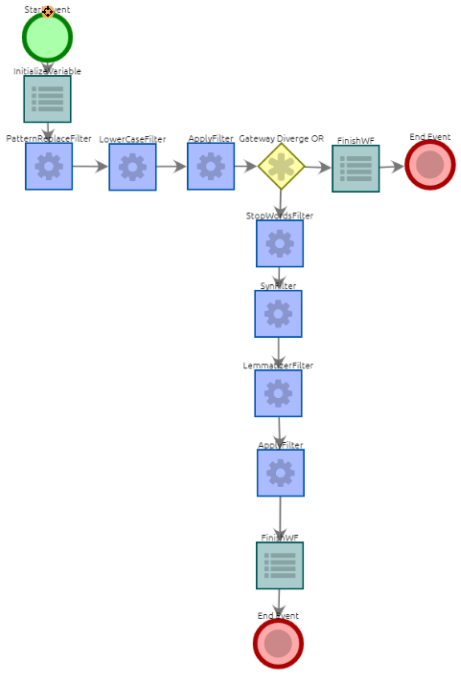

Figura 1: Pipeline generado con la herramienta Moriarty, para el pre-procesado de textos.

Mediante esta herramienta se ha desarrollado un flujo de trabajo de pre-

procesado de texto que consta de las siguientes etapas (figura 1):

- Limpieza de texto como por ejemplo URLs, emails, menciones, etc.
- Conversión a minúsculas.
- Lematización.
- Stopwords.
- Sinónimos basados en diccionarios emocionales.

2.3 Introducción de Sinónimos basados en emociones

Esta aproximación se basa en la utilización de varios diccionarios afectivos, *Dictionary of Affect in Language* construido por Cynthia M. Whissell et al. (1986) y el *Affective Norms for English Words* (ANEW; Bradley et al., 1999) en el que se incluyen un conjunto de palabras del castellano con cierta connotación afectiva y su polaridad asociada de la misma forma que fue utilizada en Del-Hoyo-Alonso et al. (2009,2015). En la última etapa del flujo de trabajo descrito en la sección previa se introducen los sinónimos de los diccionarios afectivos en los textos procesados, lo que además de reducir todavía más la dimensión del corpus permitirá al algoritmo establecer relaciones de más alto nivel.

3 Resultados

El algoritmo FastText acepta diferentes parámetros de entrada para permitir el ajuste del modelo:

- minCount: número mínimo de ocurrencias de una palabra (0-4).
- wordNgrams: máxima longitud de n-gramas (2-4).
- minn: mínimo número de carácteres del n-grama (2).
- learning rate: factor de aprendizaje (0.1-0.001)
- dim: dimensión del vector de palabras (50-200).
- ws: ventana de contexto (4-5).
- epochs: número de épocas (100-500000).
- loss: función de pérdidas (softmax).

Para la optimización de las métricas del modelo, hemos realizado una búsqueda de parámetros del algoritmo, que ha sido factible sin un coste computacional muy elevado, gracias a la alta velocidad del algoritmo.

En la tabla 1 se muestran algunos resultados obtenidos variando los parámetros del algoritmo según los valores mostrados en su definición, de los cuales, se han seleccionado los tres mejores para la competición. Principalmente se han desarrollado dos tipos de experimentos, el primero con un pre-procesamiento de diccionario afectivo y un segundo únicamente con un lematizado de palabras.

Model	Acc Desa.	M-P	M-R	M-F1	Accu.
FT-lemma	0.575	0.438	0.411	0.424	0.569
FT-lemma	0.565	0.443	0.446	0.445	0.561
FT-affective	0.585	0.450	0.423	0.436	0.576
FT-affective	**0.587**	**0.469**	**0.454**	**0.461**	**0.576**

Tabla 1: Resultados obtenidos con y sin diccionarios afectivos sobre conjunto de desarrollo.

Como se puede observar, la utilización del diccionario afectivo hace que tengamos mejores resultados, aunque el algoritmo también obtiene resultados bastante satisfactorios sin ellos. Se han presentado las métricas generales del modelo, pero cabe destacar que las clases neutras son las más difíciles de clasificar y por tanto las que aumentan el error cometido. Por ejemplo, el tweet con identificador 771115324884262912 del conjunto de desarrollo, cuyo contenido es: *"@juankipua Es que en el Ojeando el año pasado tampoco, tiene muchas canciones ya jajajajaja*" ha sido categorizado con sentimiento positivo, siendo su polaridad correcta NEUTRA. Asimismo, el tweet con identificador 771118683414560768, con contenido "*Bueno, estoy en la batalla final del Conquista y ya después me faltaría Revelación*", también ha sido categorizado con sentimiento positivo, siendo su polaridad correcta "NONE".

3.1 Comparación de resultados

El resultado final de nuestra implementación en comparación con los mejores resultados de la competición se muestra en la tabla 2.

Esperaremos a revisar los diferentes artículos del resto de grupos participantes con mayor profundidad para poder establecer comparaciones y posibles mejoras de nuestra implementación.

Group	M-P	M-R	M-F1	Accuracy
ELiRF-UPV	0.497	0.490	0.493	0.607
RETUYT	0.490	0.453	0.471	0.596
ELiRF-UPV	0.470	0.461	0.466	0.597
ITAINNOVA	**0.469**	**0.454**	**0.461**	**0.576**

Tabla 2: posición final del algoritmo en la competición (resultados sobre conjunto de test).

4 Conclusiones

Se ha presentado el algoritmo de FastText como herramienta para la mejora de la precisión del análisis de sentimiento u otros problemas de NLP donde el tamaño disponible del corpus de entrenamiento es pequeño en comparación al espacio de hipótesis. Por otra parte, se ha presentado una alternativa al Deep Learning con un resultado bastante satisfactorio. Su velocidad de computación ha hecho posible realizar la búsqueda de los parámetros del algoritmo sin disponer de una capacidad de cálculo muy elevada. A nivel general se deberá profundizar en cómo mejorar el rendimiento en las dos etiquetas neutras existentes.

Bibliografía

Bradley, M. M., & P. J. Lang. 1999. Affective norms for English words (ANEW): Instruction manual and affective ratings (pp. 1-45). Technical Report C-1, The Center for Research in Psychophysiology, University of Florida.

del-Hoyo, R., I. Hupont, F. J. Lacueva & D. Abadía. November 2009. Hybrid text affect sensing system for emotional language analysis. In Proceedings of the international workshop on affective-aware virtual agents and social robots (p. 3). ACM.

del-Hoyo-Alonso, R., M. V. Rodrigálvarez-Chamarro, J. Vea-Murguía, & R. Montañés-Salas. 2015. Ensemble algorithm with syntactical tree features to improve the opinion analysis. Comité organizador TASS 2015, 53.

Joulin, A., E. Grave, P. Bojanowski & T. Mikolov. 2017. Bag of Tricks for Efficient Text Classification. EACL 2017, 427.

Liu, P., S. Joty, & H. M. Men. September 2015. Fine-grained Opinion Mining with Recurrent Neural Networks and Word Embeddings. In EMNLP (pp. 1433-1443).

Martínez-Cámara, E., M. A. García-Cumbreras, J. Villena-Román, & J. García-Morera. 2016. TASS 2015 - The Evolution of the Spanish Opinion Mining Systems. Procesamiento del Lenguaje Natural, 56.

Martínez-Cámara, E., M. C. Díaz-Galiano, M. A. García-Cumbreras, M. García-Vega & J. Villena-Román. 2017. Overview of TASS 2017. In Proceedings of TASS 2017: Workshop on Semantic Analysis at SEPLN (TASS 2017). CEUR-WS.org, vol. 1896.

Mikolov, T., K. Chen, G. Corrado, & J. Dean- 2013. Efficient estimation of word representations in vector space. In International Conference on Learning Representations

Peña, P., R. Del-Hoyo-Alonso, J. Vea-Murgía, V.M. Rodrigálvarez, J. Calvo, & J. Martín. 2016. Moriarty: Improving 'time To Market'In Big Data And Artificial Intelligence Applications. International Journal of Design & Nature and Ecodynamics, 11(3), 230-238.

Poria, S., E. Cambria, & A. Gelbukh. 2016. Aspect extraction for opinion mining with a deep convolutional neural network. Knowledge-Based Systems, 108, 42-49.

Radford, A., R. Jozefowicz, & I. Sutskever. 2017. Learning to generate reviews and discovering sentiment. arXiv preprint arXiv:1704.01444.

Socher, R., A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, & C. Potts. October 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In Proceedings of the conference on empirical methods in natural language processing. EMNLP. Vol. 1631, p. 1642.

Villena-Román, J., J. García-Morera, M. A. García-Cumbreras, E. Martínez-Cámara, M.

T. Martín-Valdivia, L. A. Ureña-López. 2015. Overview of TASS 2015. In Proceedings of TASS 2015: Workshop on Sentiment Analysis at SEPLN. CEUR-WS.org, vol. 1397.

Whissell, C., M. Fournier, R. Pelland, D. Weir, & K. Makarec. 1986. A dictionary of affect in language: IV. Reliability, validity, and applications. Perceptual and Motor Skills, 62(3), 875-888.