Identificación de comunidades a través del lenguaje

Juan Manuel Ortiz de Zarate[1] and Esteban Feuerstein[1,2]

[1] Departamento de Computación, Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires, Argentina
[2] Fundación Sadosky, Argentina

Abstract. En este trabajo desarrollamos una metodología para identificar comunidades en redes sociales exclusivamente a través de su jerga, es decir por el lenguaje que utilizan. Presentamos resultados preliminares de nuestros experimentos en los que logramos una accuracy del 80% mediante modelos de PLN.

Keywords: PLN.Detección de comunidades.Redes Sociales

1 Introducción

La identificación de comunidades es una tarea ampliamente estudiada desde diversas disciplinas hace ya largos años. En el famoso estudio de los 70' "An information flow model for conflict and fission in small groups"[15] Zachary analiza las comunidades que se encontraban latentes dentro de un grupo de Karate basándose únicamente en la estructura del grafo generado por sus relaciones. Posteriormente, ya en épocas más recientes y con la aparición de la digitalización de las relaciones humanas se hizo mucho más accesible analizar e identificar comunidades a escalas mucho más grandes. Numerosos trabajos han propuesto algoritmos y métodos para mejorar y eficientizar estos procesos, siguiendo la idea de Zachary en cuanto a la utilización de la estructura de la red como principal soporte para luego enriquecerlo con el contenido que circula por ella [7, 12, 10, 4, 16, 11].

Hoy las redes sociales juegan un rol sumamente preponderante en nuestras interacciones, influeciándolas directamente y produciendo, debido a sus algoritmos de segregación, "Filter Bubbles"[8] que agudizan la homofilia[5]. En la práctica, se genera para cada usuario un mundo de confort en el cual todo parece coincidir con nuestros puntos de vista, opiniones y gustos. Esto produce a su vez comunidades sumamente homofílicas a su interior, es decir con usuarios que coinciden entre sí fuertemente en diversos temas.

Como ejemplo de lo anterior, en un reciente trabajo de Tran y Ostendorf [17] se analizó el lenguaje producido por distintas comunidades en *reddit*[3]. Mediante modelos de NLP pudieron identificar la jerga de distintas discusiones y clasificarlas en su respectiva comunidad con un accuracy de 86%, comprobando además que le jerga discrimina mejor que el tópico tratado. Sin embargo a la hora de clasificar un conjunto de comentarios de un mismo usuario el accuracy fue notablemente menor: 51%.

Dicho trabajo se realizó sobre comunidades no necesariamente excluyentes entre sí, por ejemplo política, noticias y ciencia. Sin embargo, desde otro enfoque, existen cuestiones particularmente controversiales donde se observan grandes polarizaciones, en las que dos comunidades representan la mayoría de los usuarios participando de forma antagónica. Ejemplos de esto son las polarizaciones políticas en distintos países, como macristas-kirchneristas en Argentina, republicanos-demócratas en Estados Unidos, a favor o en contra del impeachment en Brasil o del Brexit en Europa. Este fenómeno se ha dado en llamar la "grieta".

[3] www.reddit.com

Con el trabajo previo de Tran y Ostendorf[17] y este otro enfoque de polarizaciones, nuestra hipótesis, en este trabajo preliminar, es que en estos casos las jergas se vuelven altamente específicas y distinguibles en cada comunidad por más que hablen del mismo tema y en el mismo idioma. Entonces mediante la utilización de algoritmos de PLN, planteamos la posibilidad de predecir la pertenencia de un usuario a una comunidad únicamente por su forma de escribir en 280 caracteres, prescindiendo de la estructura del grafo y sus relaciones, logrando además una mayor eficiencia computacional en esta tarea.

Uno de los algoritmos más populares hoy en día para la identificación de comunidades es *Walktrap*[9]. Para lograr su cometido utiliza "random walks" sobre la estructura del grafo. Este método resulta muy efectivo desde un punto de vista práctico, además de funcionar bien de acuerdo a medidas generalmente aceptadas como la modularidad Q[4]. Por esto decidimos utilizarlo como ground-truth a la hora de testear la eficacia y performance de nuestros modelos. Sin embargo, el mecanismo es considerablemente ineficienciente, con una complejidad de $O(mn^2)$, donde m corresponde al número de aristas y n al de vértices. Este es uno de los factores que nos motivó a buscar otra forma de reconocer comunidades.

Para la clasificación del texto utilizamos el algoritmo desarrollado por el laboratorio de investigación de Facebook en el trabajo "Bag of Tricks for Efficient Text Classification"[2] en el cual logran equiparar el accuracy de los más modernos modelos de Deep Learning[14, 13, 1] a la vez que mejoran notablemente el costo computacional.

2 Metodología

En el proceso experimental utilizamos a *Twitter* como plataforma debido a la información de los retweets y el texto de los mismos que nos provee. En base a esto podemos graficar la red de interacciones, donde cada nodo es un usuario y las aristas representan retweets entre ellos. Las mismas son dirigidas apuntando hacia el usuario que fue retwiteado. A la vez usamos los textos introducidos en cada uno de los tweets para predecir o entrenar nuestro modelo. Dicho proceso lo dividimos en tres etapas: obtención de los tweets de cada comunidad, entrenamiento del modelo y finalmente predicción sobre nuevos casos.

Para la obtención de los tweets utilizamos dos metodologías distintas, que llamamos respectivamente *autómatica* y *manual*. La *autómatica* consiste en obtener todos los tweets relacionados a una búsqueda determinada (en nuestro caso las menciones a dos referentes políticos netamente antagonistas o la mención a algún hecho concreto y trascendente de la política) en el orden de los 400000 tweets. Luego mediante *Walktrap* identificamos dentro del grafo inducido por esa búsqueda los usuarios de las dos comunidades más grandes (que en estos casos representan más del 75% de los usuarios) y etiquetamos todos sus tweets en base a esta pertenencia. La *manual* consiste en pre-definir un conjunto de usuarios de cada comunidad con una alta tasa de actividad en la red social. Esta identificación previa la hicimos manualmente observando a qué referente de cada comunidad apoyaban cada uno de estos usuarios. Luego recolectamos todos sus tweets indiscriminadamente, es decir sin tener en cuenta sobre qué tema hablan, durante dos meses.

En la segunda etapa del proceso nos ocupamos de normalizar los tweets, pasando su codificación a ASCII minúscula y removiendo los links, menciones, signos de puntuación y caracteres de control. Posteriormente, como el clasificador es del tipo supervisado etiquetamos cada tweet en base a la comunidad de pertenencia de su creador. Este conjunto de datos lo utilizamos para generar el modelo mediante *FastText*[5].

[4] Q(G)=$\sum_{C \in G}(e_c - a_c)$, donde G es el grafo, C sus comunidades detectadas, e_c la fracción de ejes internos de la comunidad y a_c los de la frontera

[5] El entrenamiento lo realizamos con 5 epochs y un learning rate de 0.1[6]

Finalmente una vez obtenido el modelo lo probamos sobre nuevas búsquedas. Las mismas abarcan intervalos temporales diferentes, lo que nos garantiza que los tweets sean nuevos y no hayan sido usados en el entrenamiento, pero producen estructuras de grafos similares, es decir, con dos grandes comunidades polarizadas que juntas representan a la mayoría de los usuarios. Utilizando el resultado que nos da *Walktrap* como ground-truth, calculamos la matriz de confusión de las predicciones de nuestro modelo en base a los tweets. *FastText* nos provee también la probabilidad de la predicción, es decir cuan probable es, según sus cálculos, que el resultado sea el correcto. Utilizando este dato calculamos otra matriz de confusión mediante el subconjunto de aquellos tweets que predijo con una precisión mayor al 90%. En la próxima sección presentamos los resultados obtenidos en cada uno de los casos.

3 Experimentos y Resultados

Realizamos nuestros experimentos sobre tres coyunturas e idiomas distintos. Estos son: la polarización entre macristas y kirchneristas en Argentina durante el período octubre de 2017 - abril de 2018, las discusiones en torno al Impeachment de Dilma Rousseff en Brasil y la polarización entre Donald Trump y Bernie Sanders en Estados Unidos.

Para cada caso hicimos lo siguiente: dados conjuntos de entrenamiento a_1, a_2,..,a_n, anotados por comunidad en base a *Walktrap*, continuos temporalmente y no solapados, generamos modelos m_1, m_2,.., m_n respectivamente. Luego cada modelo m_i con i $\in$ (1..n) fue testeado sobre todos los a_j con j $\in$ (1..n) y i $\neq j$. Como mencionamos anteriormente calculamos la matriz de confusión y el parametro que observamos es el *balanced accuracy*[3](accuracy ponderado por cantidad de casos de cada clase) ya que no siempre se cuenta con la misma cantidad de tweets en cada comunidad.

En nuestros resultados preliminares dentro de las predicciones con una probabilidad mayor al 90% se logró en el caso de Brasil un *balanced accuracy* del 69,2% en promedio[6], de un 80% en el caso estadounidense y del 74,6% en el Argentino. En este último caso se destacó particularmente el modelo generado mediante la metodología *manual*, el cual logró un *balanced accuracy* del 81% pero sobre un subconjunto promedio del 27% (ya que no predice a todos con una probabilidad mayor al 90%). Para aquellas predicciones sin distinción de la probabilidad de acierto en el caso de Brasil se obtuvo un accuracy promedio del 64%, de un 74% en el estadounidense y del 63% en el argentino.
La generación de los modelos demora desde 10 segundos para un volumen de 140000 tweets a un minuto para 1000000 tweets, mientras que la predicción demora menos de 10 segundos para 140000 tweets. Esto logra una gran eficiencia respecto a *Walktrap* que para una red de 60000 nodos y 165000 aristas (la red promedio generada por 140000 tweets) demora varias horas. Teniendo en cuenta que la complejidad computacional de *Walktrap* es cuadrática sobre la cantidad de nodos, para redes más grandes las ventajas de nuestro método serían aún mayores, ya que su complejidad es lineal en la cantidad de tweets lo que los vuelve mucho más escalable en este sentido.

4 Conclusiones y trabajo futuro

Si bien este es un trabajo aún preliminar, los resultados obtenidos son alentadores porque nos dan la oportunidad de seguir mejorando y profundizando nuestro proceso experimental teniendo varios objetivos en el horizonte. En primer lugar lograr un modelo estándar para predecir mediante el lenguaje la pertenencia a comunidades de los usuarios de redes

[6] el promedio de los *balanced accuracy* de las predicciones de todos los modelos

similares, encontrando así una nueva forma de identificación y eficientizando el costo computacional. Para esto buscaremos utilizar otros algoritmos de identificación de comundiades de ground-truth como puede ser *K-means* y sobre otras comunidades no necesariamente políticas donde encontremos fuerte homofilia, como es el caso del deporte o los derechos humanos. También utilizaremos los word-embbeding que nos provee *FastText* para detectar mejor la semántica que diferencia a las comunidades y así agudizar el modelo y lograr una mejor generalización. Analizaremos también la posibilidad de incluir información sobre la estructura del grafo que nos mejore la performance sin incrementar demasiado el costo computacional.
Por último nos da lugar a hacernos nuevas preguntas cómo: ¿Es posible identificar la jerga de una comunidad en base a un subconjunto de usuarios influyentes en la misma? ¿Cuánto tiempo persiste la jerga que se crea en torno a una comunidad? ¿Dicha jerga es inherente únicamente a un tópico específico o es transversal a varios?¿Que eficiencia tienen estos métodos en comunidades menos polarizadas?

Bibliografía

1. Alexis Conneau, Holger Schwenk, Lo¨ıc Barrault, and Yann Lecun. 2016. Very deep convolutional networks for natural language processing. arXiv preprint arXiv:1606.01781
2. A. Joulin, E. Grave, P. Bojanowski, T. Mikolov, Bag of Tricks for Efficient Text Classification
3. BRODERSEN, Kay Henning, et al. The balanced accuracy and its posterior distribution. En Pattern recognition (ICPR), 2010 20th international conference on. IEEE, 2010. p. 3121-3124
4. LIM, Kwan Hui; DATTA, Amitava. Following the follower: detecting communities with common interests on twitter. En Proceedings of the 23rd ACM conference on Hypertext and social media. ACM, 2012. p. 317-318.
5. MCPHERSON, Miller; SMITH-LOVIN, Lynn; COOK, James M. Birds of a feather: Homophily in social networks. Annual review of sociology, 2001, vol. 27, no 1, p. 415-444.
6. HAYKIN, Simon; NETWORK, Neural. A comprehensive foundation. Neural networks, 2004, vol. 2, no 2004, p. 41.
7. OZER, Mert; KIM, Nyunsu; DAVULCU, Hasan. Community detection in political Twitter networks using Nonnegative Matrix Factorization methods. En Advances in Social Networks Analysis and Mining (ASONAM), 2016 IEEE/ACM International Conference on. IEEE, 2016. p. 81-88.
8. PARISER, Eli. The filter bubble: What the Internet is hiding from you. Penguin UK, 2011.
9. PONS, Pascal; LATAPY, Matthieu. Computing communities in large networks using random walks. En ISCIS. 2005. p. 284-293.
10. RUAN, Yiye; FUHRY, David; PARTHASARATHY, Srinivasan. Efficient community detection in large networks using content and links. En Proceedings of the 22nd international conference on World Wide Web. ACM, 2013. p. 1089-1098.
11. SACHAN, Mrinmaya, et al. Using content and interactions for discovering communities in social networks. En Proceedings of the 21st international conference on World Wide Web. ACM, 2012. p. 331-340.
12. WANG, Liaoruo, et al. Detecting community kernels in large social networks. En Data Mining (ICDM), 2011 IEEE 11th International Conference on. IEEE, 2011. p. 784-793.
13. Xiang Zhang and Yann LeCun. 2015. Text understanding from scratch. arXiv preprint arXiv:1502.01710.
14. Yoon Kim. 2014. Convolutional neural networks for sentence classification. In EMNLP.
15. ZACHARY, Wayne W. An information flow model for conflict and fission in small groups. Journal of anthropological research, 1977, vol. 33, no 4, p. 452-473.
16. ZALMOUT, Nasser. Mining the Social Web: Community Detection in Twitter, and its Application in Sentiment Analysis. 2013. Tesis Doctoral. Master's thesis, Department of Computing, Imperial College London, London, SW7 2AZ, UK.
17. TRAN, Trang; OSTENDORF, Mari. Characterizing the language of online communities and its relation to community reception. arXiv preprint arXiv:1609.04779, 2016.