Detección de discurso de odio en tweets

3 de marzo de 2022

En esta presentación se estudia la aplicación de la regresión multinomial como clasificador para el problema de detección de discurso de odio en tweets.

Discurso de odio

Aunque en la actualidad no se tiene una definición estándar de lo que constituye un discurso de odio, pues este depende usualmente del país o locación, con lo que para este trabajo definimos el discurso de odio de la siguiente manera.

Discurso de Odio

Se entiende como toda clase de comunicación verbal, escrita o comportamiento que violenta a otra persona (o personas) **en base a quiénes son**, es decir, basado en su religión, raza, etnicidad, nacionalidad, género, orientación sexual, etc.

Cabe notar que en ningún lado se dice explícitamente que el discurso de odio requiera de tener algún tipo de lenguaje ofensivo. Este punto es importante porque los modelos actuales tienen el problema de que no diferencian lenguaje ofensivo del discurso de odio, T. Davidson et al (2017)[1].

[1] Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11, 2017

Particularmente la importancia de detectar un potencial discurso de odio en las redes sociales, es porque esto puede evitar que se tenga un *crimen de odio.* Tomemos como ejemplo la siguiente noticia del New York Times. [2]

[2] Un grupo de militares subían información falsa sobre los rohingyas (población musulmana que ha vivido por varios años en Myanmar). Esto provocó que se hiciera un grupo armado que arremetió con una aldea de dicho grupo. Este tipo de discurso de odio de le conoce como *ethnic cleansing*.

De esta manera, tanto Facebook[3] como Twitter, entre otras platafor-

[3] De hecho, la misma empresa admitió que fue usado para incitar violencia, ver la noticia aquí.

mas, deben reforzar sus acciones para evitar que sucedan este tipo de situaciones.
Por otro lado, también es importante mencionar que la forma en que se detectan ciertas de manera manual es mediante el uso de monitores humanos noticia, desafortunadamente son usualmente empleos mal pagados, por tanto, el automatizar y mejorar la detección de discursos de odio (que estos pueden ser videos, texto y audio) permitiría que sea cada vez menos necesario el uso de monitores humanos.

Procesamiento del Lenguaje Natural (PLN)

Es una rama de la inteligencia artificial que permite a las computadoras a entender, interpretar y manipular el lenguaje humano. De forma general un problema de clasificación en PLN sigue el siguiente esquema.

1. Entrada: Texto.
2. Representación del texto.
3. Clasificador
4. Salida: Clase

Datos

Para el texto usaremos la base de datos de T. Davidson et al [4], donde tomaron una muestra de aproximadamente 25,000 tweets, que contenían los términos del lexicon Hatebase[5], esto es, una lista de palabras que distintas comunidades han acordado conjuntamente que se usan en discursos de odio, tal lista continua actualizándose hasta el día de hoy. Después dichos tweets fueron anotados manualmente mediante crowdsourcing. Para este trabajo sólo usaremos los tweets y la clase asignada a cada uno, ver Figura 1, donde se usó la siguiente nomenclatura.

- 0: Hate Speech,
- 1: Offensive Language,
- 2: Neither.

[4] Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11, 2017

[5] `https://hatebase.org/`

class	tweet
2	!!! RT @mayasolovely: As a woman you shouldn't...
1	!!!!! RT @mleew17: boy dats cold...tyga dwn ba...
1	!!!!!!! RT @UrKindOfBrand Dawg!!!! RT @80sbaby...
1	!!!!!!!! RT @C_G_Anderson: @viva_based she lo...
1	!!!!!!!!!!! RT @ShenikaRoberts: The shit you...
1	!!!!!!!!!!!!!!!"@T_Madison_x: The shit just...
1	!!!!!!"@__BrighterDays: I can not just sit up ...

Figura 1: Captura de la base de datos.

T. Davidson et al [6], descubrieron que el discriminar entre discurso de odio y lenguaje ofensivo es bastante complicado, en principio, es de notar que los anotadores consideraron asignar la mayoría de los tweets como lenguaje ofensivo, y una muy pequeña cantidad, como discurso de odio. Es decir, la distribución de clases tiene ciertamente un sesgo, ver Figura 2. Más aún, obtuvieron las siguientes conclusiones,

- Los anotadores consideran términos homofóbicos o racistas como lenguaje de odio. Pero, tweets sexistas o derogatorios hacía mujeres sólo ofensivos.
- Si los tweets no contienen groserías entonces tienden a ser clasificados como *neither.*
- Existe un fenómeno interesante en algunos tweets, esto es, cuando por ejemplo, se contesta un tweet racista con uno homofóbico. Sin duda dicho tweet debe ser considerado como discurso de odio, pero la parte de clasificación puede tener problemas.

Veremos más adelante si podemos concluir lo mismo. Este tipo de problemáticas han tomado gran relevancia en la actualidad que se han organizado conferencias respecto al tema, se tiene por ejemplo, el congreso de abuso y daños (Abuse and Harms[7]) que recientemente (2020) tuvo su cuarta edición, donde se trataron temas como detección de lenguaje ofensivo en plataformas digitales, los sesgos inherentes que las bases de datos [8], así como estudiar el discurso de odio particularmente hacía personas asíaticas-americanas, y en temas más actuales, los discursos de odio por COVID-19.

[6] Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11, 2017

[7] https://www.workshopononlineabuse.com/past-workshops/woah-2020-home

[8] Thomas Davidson, Debasmita Bhattacharya, and Ingmar Weber. Racial bias in hate speech and abusive language detection datasets. *CoRR*, abs/1905.12516, 2019. URL `http://arxiv.org/abs/1905.12516`

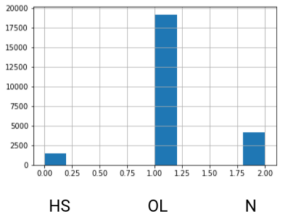


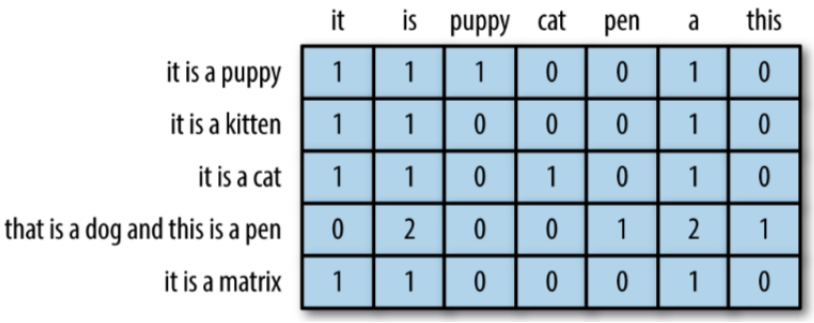
Figura 2: Distribución de las clases de los tweets

Representación del Texto

Una vez considerados estos tweets, requerimos poder representarlos en objetos cuantitativos que permitan ser manipulables y a la vez, contengan información relevante del tweet, a esta información en PLN se le llaman *features.*

En concreto, vamos a considerar los siguientes *features*:

1. **Bolsa de palabras**. Extraemos todos los tokens (palabras o caracteres) de cada tweet para poder construir una matriz de término-documento, esta en su forma más sencilla (de pesado binario) determina si una palabra (columna) está presente o no en un tweet (renglón). Para este trabajo, usamos un pesado un poco más detallado, denominado TF-IDF, que toma en cuenta no nada más el número de veces (TF) que cierto token aparece en el documento (tweet) , sino que también qué tan raro o poco común es en otros documentos (IDF).



	it	is	puppy	cat	pen	a	this
it is a puppy	1	1	1	0	0	1	0
it is a kitten	1	1	0	0	0	1	0
it is a cat	1	1	0	1	0	1	0
that is a dog and this is a pen	0	2	0	0	1	2	1
it is a matrix	1	1	0	0	0	1	0

Figura 3: Matriz término-documento

2. **Análisis de Sentimiento.** Asignarle un sentimiento a cada tweet, este es una probabilidad de que el tweet sea clasificado como Positivo, Negativo o Neutro[9]

3. **Part of Speech**. Para cada palabra, requerimos considerar etiquetas correspondientes a si es un adverbio, un número, un símbolo, un adjetivo, etc. dependiendo del rol que tiene en cada tweet. También se tiene una matriz documento-término[10]

4. **Flesch-Kincaid-Test** Existe también un pesado que permite determinar qué tan legible es un tweet, es decir, si un niño de 12 o un adulto mayor puede comprender dicho texto[11].

5. **Metadatos.** Finalmente, también consideramos el número de hashtags, de menciones, de retweets, de palabras, de sílabas, el porcentaje de símbolos, etc.

[9] https://towardsdatascience.com/fine-grained-sentiment-analysis-in-python-part-1-2

[10] https://medium.com/analytics-vidhya/getting-started-with-nlp-tokenization-document-ter

[11] https://en.wikipedia.org/wiki/Flesch-Kincaid_readability_tests

Modelo

Regresión Multinomial.

El método por elección en la literatura para el clasificador a usar es la regresión logística, pues requerimos determinar si un tweet es o no discurso de odio. En términos de probabilidad, deseamos calcular

$$\mathbb{P}\left(y = c|x\right), \quad c \in \{HS, OL, N\}, x \text{ un tweet.}$$

Para ello usamos la generalización de la regresión logística, esto es, la **regresión multinomial**, una gran ventaja de esta herramienta es que no requiere de hipótesis de normalidad, linealidad ni homoestaticidad.
Suponemos que tenemos C clases, y a cada tiene una le corresponde una relación lineal

$$E[y] = f_c(\mathbf{x}, \boldsymbol{\theta}^{(c)}) = \sum_{d=1}^{D} \theta_d^{(c)} x_d + \theta_0^{(c)}, \quad c = 1, 2, \ldots, C. \tag{1}$$

Como deseamos calcular $p(y|x)$, cuando y puede tomar valores en $\{1, 2, \ldots, C\}$, para ello hacemos uso de la función *softmax*, esto es,

$$\mathbb{P}\left(y = c|\boldsymbol{x}\right) = \frac{\exp\left(f_c(\mathbf{x}, \boldsymbol{\theta}^{(c)})\right)}{\sum_{k=1}^{C} \exp\left(f_k(\mathbf{x}, \boldsymbol{\theta}^{(k)})\right)}.$$

Notemos que el número de parámetros a ser estimados es $C \times D$, pues cada clase tiene su propio vector de parámetros $\boldsymbol{\theta}^c$. Usando los datos de entrenamiento, los pesos se pueden estimar maximizando la función de máxima verosimilitud

$$\mathbb{P}\left(\boldsymbol{y}|\boldsymbol{x}; \boldsymbol{\theta}\right) = \prod_{n=1}^{N} \prod_{c=1}^{C} \left(p_n^{(c)}\right)^{y_n^c}.$$

Donde

$$p_n^{(c)} = \mathbb{P}\left(y = c|\boldsymbol{x}_n\right) = \mathbb{P}\left(y_n^{(c)} = 1|\boldsymbol{x}_n; \Theta\right) = \frac{\exp\left(\boldsymbol{x}_n^T \boldsymbol{\theta}^c\right)}{\sum_{k=1}^{C} \exp\left(\boldsymbol{x}_n^T \boldsymbol{\theta}^{(c)}\right)}.$$

Clasificador

Los ingredientes de un clasificador son:

1. Un conjunto de datos $(\vec{x}_i, y_i)_{i=1}^n$.
2. Una función de clasificación que permita determinar $\hat{y}$, la clase estimada. En nuestro caso representa determinar

$$\mathbb{P}(y = c|x) = \text{softmax}(x).$$

3. Una función de aprendizaje, en nuestro caso es la función de entropía cruzada.

$$L_{CE}(\hat{y}, y) = -\sum_{k=1}^{K} 1_{\{y=k\}} \log \mathbf{P}(y = k|x).$$

4. Algún algoritmo de optimización que permita determinar los puntos críticos de L_{CE}.

En la fase entrenamiento obtenemos los pesos $\hat{\theta}$, mediante la solución numérica del problema $\text{mín}_\theta\, L_{CE}$, tomando un subconjunto de los datos $(\mathbf{x}_i, y_i)_{i=1}^N$, con $N < n$. Luego en la fase de prueba, usamos el resto de los datos para clasificarlos usando los pesos determinados en el paso anterior, calculando $\text{máx}_c\, \mathbb{P}_{\hat{\theta}}(y = c|x)$.

Selección de variables

Por otro lado, como la matriz de covariables es una matriz rala, T. Davidson et al (2017) proponen usar el método de regresión con penalización o con término de regularizado, principalmente, porque

permite reducir la dimensión de esta, y sólo quedarnos con las columnas que sean más relevantes para el problema de clasificación.

Least Absolute Shrinkage and Selection Operator[12] **(LASSO)** Para términos interpretativos, sería deseable no nada más hacer cada vez más pequeños los pesos $\boldsymbol{\theta}$, sino también tener la posibilidad de que sean exactamente cero. Para ello, agregamos el término de penalización siguiente

$$\hat{\boldsymbol{\theta}}_{\text{LASSO}} = \arg\max_{\theta} \left(\mathbb{P}\left(\boldsymbol{y}|\boldsymbol{x};\boldsymbol{\theta}\right) + \boxed{\sum_{d=1}^{D} \|\boldsymbol{\theta}^{c}\|} \right).$$

Como el estimador $\hat{\boldsymbol{\theta}}_{\text{LASSO}}$ queda definido en términos de la norma de los pesos, para pesos muy grandes, el término de penalización crece lentamente, pero se mueve lejos de cero más rápido para coeficientes más cercanos a cero, consecuentemente, los coeficientes pequeños son presionados hacía el cero, mientras que los coeficientes grandes no se verán afectados por esta penalización[13] , ver Figura 4.

[12] Ludwig Fahrmeir, Thomas Kneib, Stefan Lang, and Brian Marx. *Regression*. Springer, 2007

[13] Ludwig Fahrmeir, Thomas Kneib, Stefan Lang, and Brian Marx. *Regression*. Springer, 2007

Resultados

Una vez hecha la reducción de dimensionalidad mediante LASSO, usamos una regresión multinomial. Tomando el 90 % de los *features* para entrenamiento, y 10 % para pruebas, obtuvimos los siguientes resultados.

En la Figura 5, de la matriz de confusión observamos que mientras la clasificación fue buena para las clases de ninguno y lenguaje ofensivo, es decir, hubo muy pocos falsos positivos. Seguimos teniendo un muy bajo valor para el discurso de odio, pues obtuvimos que el 77 % de las veces se equivoca, y en particular, el 65 % clasifica el discurso de odio como lenguaje ofensivo.

Es importante también mirar a los triángulos formados en la diagonal, pues notemos que el que se encuentra en la superior derecha, significa el porcentaje de veces que no pudimos diferenciar lenguaje ofensivo de discursos de odio, mientras que el triángulo inferior izquierdo, es el porcentaje de veces que pudimos haber llamado a alguien una persona que perpetúa el discurso de odio cuando no es así.

Si miramos con más detalle a los tweets que no fueron clasificados correctamente, notamos que ciertamente se confirman las aseveraciones de T. Davidson et al (2017)[14], pues notamos que tweets que contienen ciertas palabras son clasificados como discurso de odio, estas son: *queer* y *bitch*, y algunos tweets que claramente muestran sexismo, que los anotadores clasificaron como ninguno, tienden a ser clasificados como lenguaje ofensivo. Ver Figura 6.

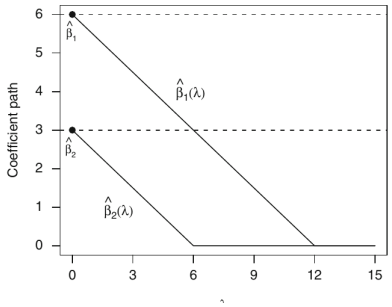


Figura 4: Comportamiento de los coeficientes para el caso $D = 2$.

	precision	recall	f1-score	support
0	0.31	0.24	0.27	164
1	0.88	0.91	0.90	1905
2	0.72	0.68	0.70	410
accuracy			0.83	2479
macro avg	0.64	0.61	0.62	2479
weighted avg	0.82	0.83	0.82	2479

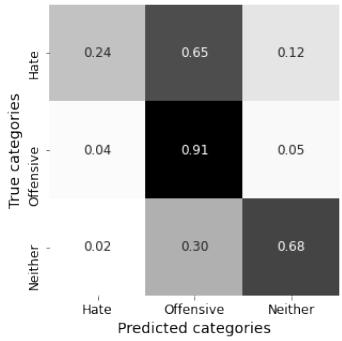
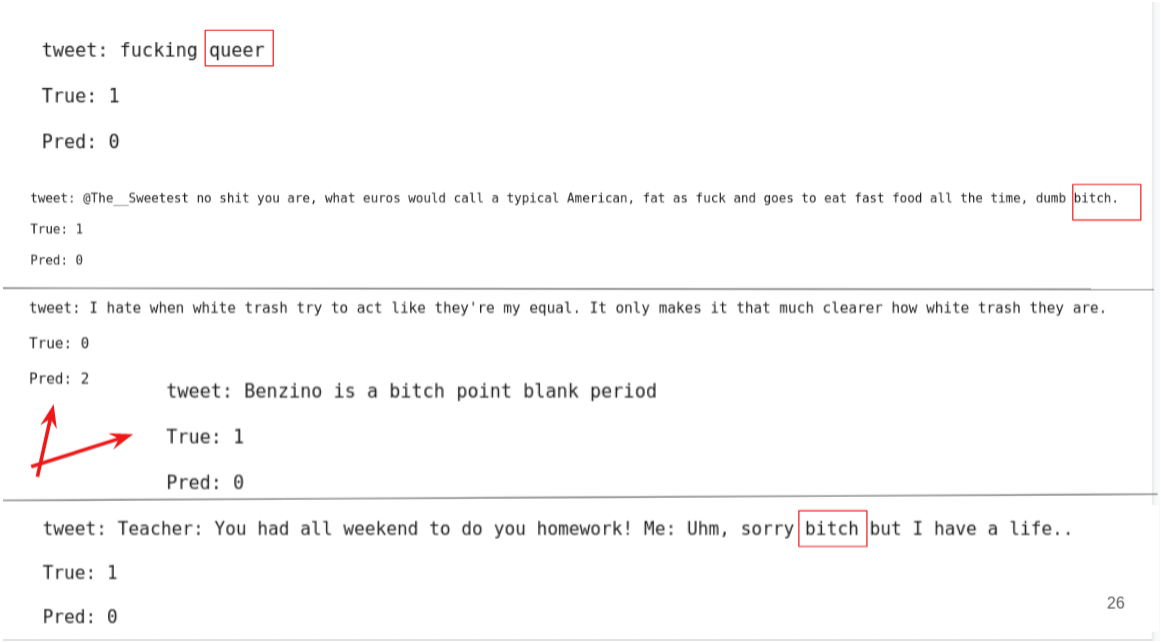


Figura 5: Reporte de métricas, y matriz de confusión.

[14] Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11, 2017

También observamos el sesgo de los anotadores respecto a no diferenciar entre discurso de odio y discurso de odio. Además, ciertas palabras hacen que el clasificador se equivoque más. Estas son *f*ggot*, *n*igga.*

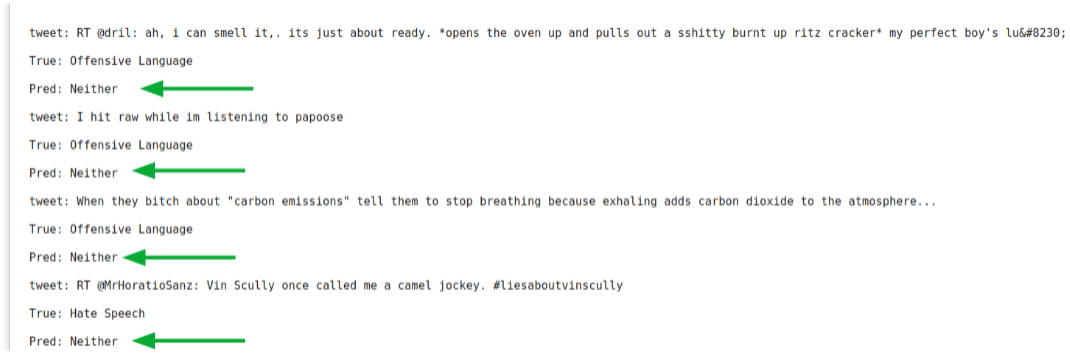


Estado del arte

Finalizamos mencionando que en la actualidad la detección de discurso de odio se ha hecho mediante modelos de aprendizaje máquina profundo, a diferencia de la estrategia del clasificador donde se requirieron de dos pasos (determinar una representación del texto para alimentar al clasificador), en aprendizaje profundo, los modelos son directamente alimentados con el texto[15]. Más aún, como estos modelos usualmente están pre entrenados con textos más generales (por ejemplo, BERT con textos de Wikipedia), permite que el clasificador al menos pueda distinguir tweets no ofensivos.
Respecto a usar métodos de aprendizaje máquina ó aprendizaje máquina profundo, Gröndahl et al. (2018)[16] probaron distintos modelos desde modelos lineales hasta de aprendizaje profundo actuales, y no hubo gran diferencia, esto es porque la efectividad del modelo depende más de la cantidad disponible de datos (tweets) y el método de anotación que se usó.

[15] Marzieh Mozafari, Reza Farahbakhsh, and Noel Crespi. A bert-based transfer learning approach for hate speech detection in online social media. In *International Conference on Complex Networks and Their Applications*, pages 928–940. Springer, 2019

[16] Tommi Gröndahl, Luca Pajola, Mika Juuti, Mauro Conti, and N Asokan. All you need is" love" evading hate speech detection. In *Proceedings of the 11th ACM Workshop on Artificial Intelligence and Security*, pages 2–12, 2018



Conclusiones

Descubrimos que independientemente del modelo, la detección de discurso de odio sigue siendo un tema complicado, y por tanto, aún hay mucho por hacer. En el caso del idioma español los modelos actuales tienen un *accuracy* de a los más el 75 %, esto es a causa de la falta de datos que se tienen. En cuanto al sesgo inherente en las bases de datos Davidson et al (2019)[17], mediante pruebas de hipótesis lograron identificar que ciertas bases de datos ocupadas como *baseline* en diversos artículos están sesgadas, esto usualmente es a causa de la elección del protocolo de anotación, es decir, quién y cómo se va a llevar a cabo la clasificación manual de los tweets.

[17] Thomas Davidson, Debasmita Bhattacharya, and Ingmar Weber. Racial bias in hate speech and abusive language detection datasets. *CoRR*, abs/1905.12516, 2019. URL `http://arxiv.org/abs/1905.12516`

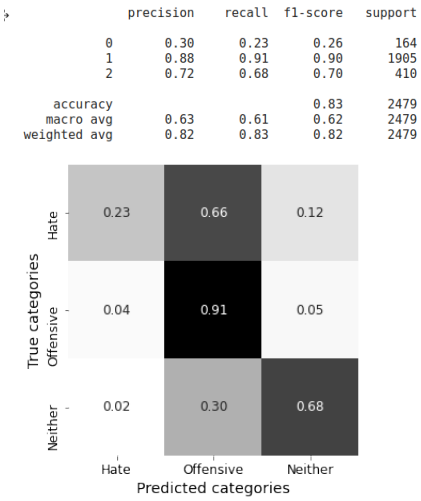


Figura 7: Reporte de métricas, y matriz de confusión. **Sin LASSO**.

Referencias

Aymé Arango, Jorge Pérez, and Barbara Poblete. Hate speech detection is not as easy as you may think: A closer look at model validation (extended version). *Information Systems*, page 101584, 2020.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11, 2017.

Thomas Davidson, Debasmita Bhattacharya, and Ingmar Weber. Racial bias in hate speech and abusive language detection datasets. *CoRR*, abs/1905.12516, 2019. URL `http://arxiv.org/abs/1905.12516`.

Ludwig Fahrmeir, Thomas Kneib, Stefan Lang, and Brian Marx. *Regression*. Springer, 2007.

Purnama Sari Br Ginting, Budhi Irawan, and Casi Setianingsih. Hate speech detection on twitter using multinomial logistic regression

classification method. In *2019 IEEE International Conference on Internet of Things and Intelligence System (IoTaIS)*, pages 105–111. IEEE, 2019.

Tommi Gröndahl, Luca Pajola, Mika Juuti, Mauro Conti, and N Asokan. All you need is"love.^{ev}ading hate speech detection. In *Proceedings of the 11th ACM Workshop on Artificial Intelligence and Security*, pages 2–12, 2018.

Xiaolei Huang, Linzi Xing, Franck Dernoncourt, and Michael J Paul. Multilingual twitter corpus and baselines for evaluating demographic bias in hate speech recognition. *arXiv preprint arXiv:2002.10361*, 2020.

Sean MacAvaney, Hao-Ren Yao, Eugene Yang, Katina Russell, Nazli Goharian, and Ophir Frieder. Hate speech detection: Challenges and solutions. *PloS one*, 14(8):e0221152, 2019.

Marzieh Mozafari, Reza Farahbakhsh, and Noel Crespi. A bert-based transfer learning approach for hate speech detection in online social media. In *International Conference on Complex Networks and Their Applications*, pages 928–940. Springer, 2019.

Flor Miriam Plaza-del Arco, M Dolores Molina-González, L Alfonso Ureña-López, and M Teresa Martín-Valdivia. Comparing pretrained language models for spanish hate speech detection. *Expert Systems with Applications*, 166:114120, 2021.

Clément W Royer, Michael O'Neill, and Stephen J Wright. A newton-cg algorithm with complexity guarantees for smooth unconstrained optimization. *Mathematical Programming*, 180(1):451–488, 2020.