Università degli studi di
Milano-Bicocca

Advanced Machine Learning

Final Project

Toxic Comment Classification

Authors:
Roberta Bensi - 861555 - r.bensi@campus.unimib.it
Martina Roberta Cecchetto - 852566 - m.cecchetto2@campus.unimib.it
Giuseppe Monea - 850432 - g.monea@campus.unimib.it

29 gennaio 2021

Sommario

Nel seguente report viene svolto un task di classificazione testuale sul dataset *Toxic Comment Classification* fornito da Kaggle. Nella fase di preprocessing per trattare il problema del dataset sbilanciato sono state utilizzate tecniche di data augmentation e subsampling. Per la risoluzione del task sono stati seguiti due principali approcci metodologici. Il primo consiste in una classificazione multilabel su tutte le classi, mentre, il secondo è composto da più step. In primo luogo si effettua una classificazione binaria per determinare la tossicità o meno del commento e solo successivamente si applica una seconda classificazione per determinare il tipo di tossicità ad esso correlato. Le reti neurali prese in considerazione per addestrare i classificatori sono la LSTM, la GRU e la CNN. Infine, sulla base di metriche come F1-score e recall si è valutata come migliore l'architettura LSTM applicata al secondo approccio.

1 Introduzione

Negli ultimi decenni la possibilità di dialogare e discutere online con persone di tutte le parti del globo ha reso più facile imbattersi in utenti tossici il cui comportamento contribuisce a peggiorare il clima, seppur virtuale, dell'ambiente in cui si trovano. Allo scopo di contenere il fenomeno della tossicità online partecipa attivamente il "Conversation AI Team", un'iniziativa di ricerca fondata da Jigsaw e Google al fine di migliorare le conversazioni online. L'obiettivo dell'analisi è pertanto quello di trovare un modello il più preciso possibile nell'individuare la tossicità nei commenti. Questi ultimi oltre a poter essere classificati come tossici sono anche differenziati in base alla loro appartenenza a diverse categorie di tossicità: severamente tossico, osceno, minaccia, insulto, odio rivolto all'identità. Al fine di individuare i commenti in questione, dopo una fase di preprocessamento in cui si sono manipolati i commenti per poterli analizzare tramite delle reti neurali, è stato necessario effettuare un ribilanciamento del dataset poiché presentava un numero di gran lunga maggiore di commenti non offensivi.
Successivamente si è potuto proseguire nella creazione dei modelli di classificazione tramite l'utilizzo di reti neurali in grado di effettuare l'analisi sequenziale delle parole presenti nei commenti. A tal fine si è deciso di sfruttare due modelli di "Recurrent Neural Network" e uno di "Convolutional Neural

Network". Il primo modello RNN presenta architettura LSTM ("Long short-term memory") e il secondo una loro variante più recente, la GRU ("Gated Recurrant Unit"). Il modello CNN presenta invece dei livelli convoluzionali a una dimensione, utili per l'analisi testuale.
Successivamente si è deciso di effettuare un'ulteriore passaggio sfruttando la rete che aveva ottenuto i risultati migliori, per capire se fosse più conveniente un'analisi separata per individuare prima l'eventuale tossicità e, solo in seguito, partendo dal presupposto che un commento fosse già categorizzato come tossico, la sua tipologia.

2 Datasets

I dati utilizzati, forniti dalla competizione Toxic Comment Classification su Kaggle, raccolgono da Wikipedia 312735 commenti in lingua inglese e il loro tipo di tossicità. La valutazione dei commenti è avvenuta grazie alla partecipazione di 5000 individui a cui è stato chiesto di votare il livello di tossicità di ogni commento, definendolo tossico se induce gli interlocutori che lo leggono a non partecipare più alla discussione. Per ogni frase vengono inoltre identificate 5 sottocategorie: severamente tossico, osceno, minaccia, insulto, odio rivolto all'identità. I dataset che sono stati utilizzati sono 3: "*train*", "*test*" e "*test_labels*". I commenti totali sono divisi in due dataset: 159571 nel train e 153164 nel test. Nel train sono presenti le seguenti 8 colonne:

- id: identificatore univoco del commento;
- comment_text: testo del commento;
- toxic: 1 se il commento è stato classificato come tossico, 0 altrimenti;
- severe_toxic: 1 se il commento è stato classificato come severamente tossico, 0 altrimenti;
- obscene: 1 se il commento presenta delle oscenità, 0 altrimenti;
- threat: 1 se il commento presenta delle minacce, 0 altrimenti;
- insult: 1 se nel commento sono stati rilevati degli insulti, 0 altrimenti;
- identity_hate: 1 se il commento presenta un odio rivolto all'identità della persona (ad es. credo religioso, orientamento sessuale, etc), 0 altrimenti.

In *test* sono presenti le colonne id e comment_text, mentre in *test_labels* le colonne id, toxic, severe_toxic, obscene, threat, insult, identity_hate.

2.1 Data Preprocessing

Non tutti i 153164 commenti presenti in *test* vengono utilizzati in quanto alcune righe del dataset *test_labels* presentano "-1" come valore di ogni classe. Queste righe e il loro corrispondente commento nel test set sono stati eliminati, rendendo 63978 i commenti utilizzabili in fase di test.
Dopo una prima esplorazione del dataset *train* ci si è accorti che 931 commenti assumevano "0" in tossicità sebbene fosse presente "1" in almeno una delle sottocategorie, per questo motivo è stata creata una funzione che in questi casi converte lo 0 in 1 nella colonna *toxic*.
Dopo questa modifica è stata effettuata una pulizia del testo. Inizialmente le frasi sono state trasformate tutte in minuscolo, dopodiché sono state estese le parole contratte (ad es. "don't" è stato trasformato "do not", "i'm" in "i am"). Successivamente è stato rimosso tutto ciò che non fosse parola (punteggiatura, simboli, etc) e infine sono state rimosse le *stop words*. In un primo momento si è inoltre provata la tecnica di *lemmatization*, tuttavia non portando miglioramenti nei risultati si è deciso di procedere senza.
Una volta realizzate queste modifiche è stata effettuata la *tokenizzazione* dei commenti impostando a 60000 il numero totale di parole da tenere come vocabolario in quanto più frequenti. Infine si è scelto 150 come valore per la lunghezza di ogni commento, troncando quindi commenti più lunghi e utilizzando la tecnica del *padding* per i commenti più corti.

2.2 Problema delle classi sbilanciate

Effettuando l'esplorazione del dataset è facilmente identificabile un problema di sbilanciamento. Infatti il 90% delle osservazioni appartiene alla classe *non toxic* (0), mentre solo il restante 10% appartiene alla classe *toxic* (1). Lo sbilanciamento delle due classi, se non corretto, porta a una bassa capacità del modello di distinguere la classe rara. Per ovviare al problema si sono tentati gli approcci di *data augmentation* e *subsampling*.
Per quest'ultimo si sono selezionate tutte le osservazioni che contenevano commenti tossici con le annesse sottocategorie di tossicità e successivamente si è effettuato un campionamento casuale dei commenti non tossici per estrarre lo stesso numero di osservazioni. Il risultato è quindi un dataset bilanciato di circa 30000 osservazioni.
Per la data augmentation invece si è partiti dal dataset originario cercando di aumentare la numerosità dei commenti *toxic* utilizzando i metodi di *Synonym*

Replacement, *Random Swap*, *Random Deletion*. Per le diverse sottoclassi di tossicità sono state applicate più volte le tecniche di data augmentation, partendo ogni volta dal dataset precedentemente aumentato, poiché le classi risultavano fortemente rare (Tabella 1).

Tabella 1: Numero di osservazioni prima e dopo la Data Augmentation

Label	**Prima**	**Dopo**
toxic	16225	109108
severe_toxic	1595	27584
obscene	8449	67573
threat	478	21010
insult	7877	65339
identity_hate	1405	24610

3 Approccio Metodologico

Ottenuti i dataset bilanciati si sono potuti definire i modelli con cui classificare i commenti. Essi sono stati utilizzati inizialmente per stabilire tramite una classificazione multi-label tutte le classi di ogni commento, considerando la classe *toxic* equivalente alle altre e non come fattore discriminante per la successiva sottocategorizzazione. La rete neurale che ha ottenuto i risultati migliori con questo approccio è stata poi utilizzata per effettuare un'ulteriore analisi. In particolare si è voluto testare se fosse più performante effettuare preventivamente il riconoscimento dell'appartenenza alla categoria *toxic* e, se questa condizione veniva verificata, riconoscere l'appartenenza alle diverse tipologie di tossicità. Per entrambi gli approcci e per tutti i modelli è stata implementata la tecnica dell'*early stopping* durante il training per monitorare l'andamento della *validation accuracy*. Per ogni modello è stato scelto come ottimizzatore *Adamax*, una variante di *Adam* particolarmente efficiente per modelli con embedding, con un *learning rate* di 0,001 per il dataset con subsampling e di 0,0001 nel caso di data augmentation.

3.1 Classificazione Multi-label

Il primo approccio ha previsto la classificazione multi-label tramite tre reti neurali: Long short-term memory (LSTM), Gated Recurrant Unit (GRU) e Convolutional Neural Network (CNN).

3.1.1 Long short-term memory

La "Long Short-Term Memory" (LSTM) è un'architettura di rete neurale di tipo "Recurrent Neural Network" (RNN). Si è deciso di utilizzare reti ricorrenti per dare risalto alla natura sequenziale che è intrinseca nella scrittura. In particolare si è deciso di considerare le RNN bidirezionali, per tenere conto in modo più preciso della posizione delle parole non solo rispetto a quelle precedenti, ma anche rispetto alle successive. In molti casi, infatti, determinate parole assumono significato differente se seguite da termini diversi. La LSTM, rispetto alle più semplici RNN, è stata selezionata per la sua peculiarità di tenere in considerazione più efficacemente dei periodi (nel caso in analisi le parole) più lontani rispetto al dato sotto osservazione.

La struttura della rete LSTM, visibile in Tabella 2 prevede inizialmente uno strato di *embedding* che ci permette di ottenere una rappresentazione distribuita delle parole e del loro relativo significato.

Tabella 2: Struttura LSTM

Layer	**Dimensione**	**# Parametri**
Embedding	60000x128	7680000
Bidirectional LSTM (dropout=0.2)	128	263168
Dense	6	1542
# Parametri		7944710

3.1.2 Gated Recurrant Unit

La Gated Recurrent Unit (GRU) è una variante della LSTM che mantiene la memoria di breve-lungo termine riducendo il numero di parametri e la complessità architetturale. Essa utilizza un approccio più snello per le dipendenze di lungo termine, conglobando in un unico gate il flusso di aggiornamento e propagazione delle informazioni. La struttura della rete GRU è visualizzata in Tabella 3.

Tabella 3: Struttura GRU

Layer	**Dimensione**	**# Parametri**
Embedding	60000x64	3840000
Bidirectional GRU (dropout=0.2)	128	148992
Dense	6	1542
# Parametri		3990534

3.1.3 Convolutional Neural Network

La Convolutional Neural Network (CNN) è un tipo di rete neurale feed forward. Originariamente creata per il riconoscimento di immagini e video riscontra buoni risultati anche nei campi della classificazione del testo e in problemi NLP (natural language processing). I filtri convoluzionali a una dimensione funzionano come *n-grams detectors* individuando le associazioni tra parole, mentre gli strati di max pooling estraggono gli n-grams rilevanti per il problema a cui la rete neurale è sottoposta. La struttura della rete neurale convoluzionale è descritta in Tabella 4.

Tabella 4: Struttura CNN

Layer	**Dimensione**	**# Parametri**
Embedding	60000x124	7440000
Conv1D (activation=relu)	8x64	63552
MaxPooling1D	4	
Conv1d (activation=relu)	8x32	16416
MaxPooling1D	4	
Flatten	224	
Dense (activation=relu)	16	3600
Dropout (rate=0.2)		
Dense (activation=sigmoid)	6	102
# Parametri		7523670

3.2 Classificazione binaria della tossicità e classificazione multi-label del tipo di tossicità

Il secondo approccio è stato strutturato dividendo il problema principale di classificazione multi-label in due sottoproblemi. In primo luogo si è impostato un problema di classificazione binaria per ogni commento, al fine di stabilire la sua appartenenza alla classe *toxic*. La rete neurale è stata scelta, invece, confrontando le metriche della strategia multi-label ed è stata selezionata l'architettura LSTM. In secondo luogo si è classificata la tipologia di tossicità dei commenti appartenenti alla classe *toxic*. Per questo scopo è stato quindi utilizzato un dataset con i soli commenti segnalati come tossici. Prima di effettuare la classificazione si è tuttavia eseguita nuovamente la fase di tokenizzazione, così da aggiungere al vocabolario solamente parole provenienti dai commenti tossici.

3.2.1 Modello di classificazione binaria

Per svolgere questo problema di clasificazione binaria si è utilizzata la rete LSTM visibile in Tabella 5.

Tabella 5: Struttura LSTM

Layer	**Dimensione**	**# Parametri**
Embedding	60000x128	7680000
Bidirectional LSTM (dropout=0.2)	128	263168
Dense	1	257
# Parametri		7943425

3.2.2 Modello di classificazione multi-label del tipo di tossicità

Per classificare il tipo di tossicità si è utilizzata anche in questo caso una rete LSTM la cui struttura è mostrata in Tabella 6.

Tabella 6: Struttura LSTM

Layer	**Dimensione**	**# Parametri**
Embedding	60000x128	7680000
Bidirectional LSTM (dropout=0.3)	128	263168
Dense	5	1285
# Parametri		7944453

4 Risultati e Valutazioni

L'obiettivo dello studio è quello di trovare un modello adatto a un problema di classificazione binaria multi-label e, per questo motivo, si è utilizzata come funzione obiettivo la *binary crossentropy*. Per valutare le performance dei vari modelli si è inoltre deciso di utilizzare come metriche la *precision*, la *recall* e l'AUC (area under the ROC curve). In aggiunta si sono confrontati i valori predetti con quelli forniti dal test-set per verificare il comportamento del modello su commenti non utilizzati in fase di training. Per questo scopo sono state utilizzate varie misure: *precision*, *recall* e *F1-score* che risulta particolarmente indicato in caso di classi sbilanciate.

4.1 Risultati Classificazione Multi-label

Il processo di training e validazione è mostrato nelle Figure 1,2,3 e 4. Avendo ottenuto prestazioni sempre superiori con la tecnica di data augmentation si è deciso di non mostrare i risultati ottenuti tramite subsampling.

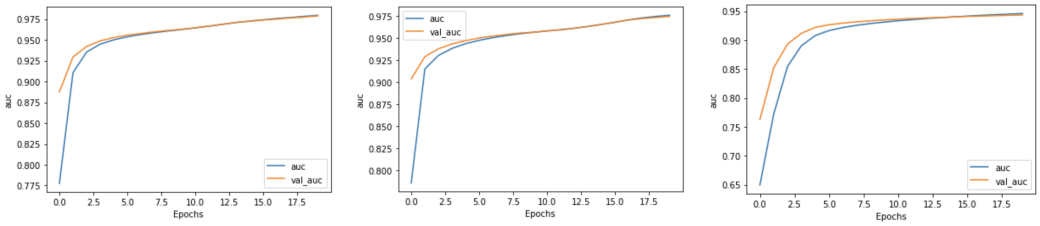

Figura 1: AUC: LSTM, GRU, CNN

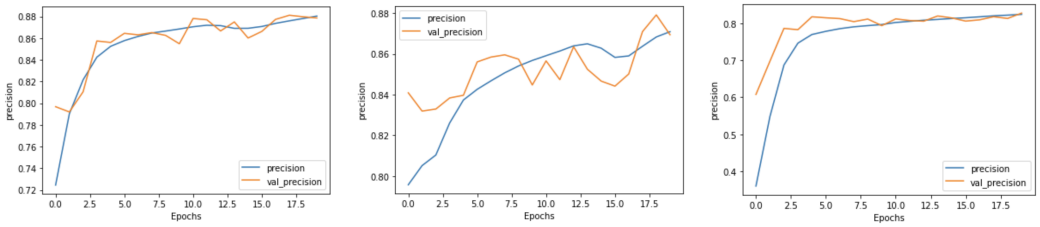

Figura 2: Precision: LSTM, GRU, CNN

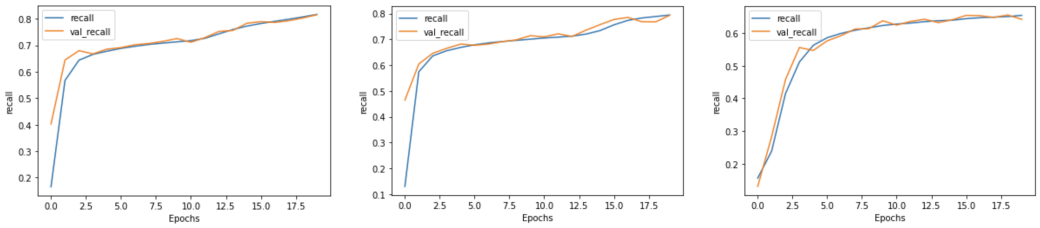

Figura 3: Recall: LSTM, GRU, CNN

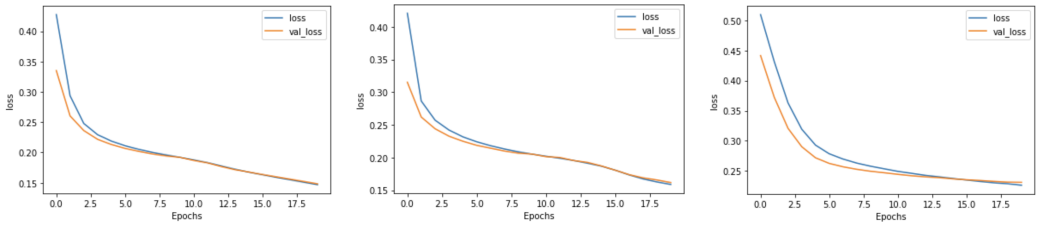

Figura 4: Loss: LSTM, GRU, CNN

Di seguito (Tabella 7) si sono calcolate le metriche relative al confronto tra predizioni e valori reali del test-set.

Tabella 7: Valutazione dei modelli tramite test set su un totale di 63978 commenti.

	LSTM			**GRU**			**CNN**			
	precision	recall	f1 score	precision	recall	F1 score	precision	recall	F1 score	support
toxic										6090
subsampling	0.39	0.93	0.55	0.38	0.93	0.54	0.25	0.83	0.39	
data augmentation	0.42	0.92	0.57	0.44	0.91	0.59	0.30	0.78	0.43	
severe toxic										367
subsampling	0.44	0.26	0.32	0.40	0.27	0.32	0.49	0.17	0.25	
data augmentation	0.27	0.68	0.39	0.29	0.62	0.40	0.29	0.53	0.38	
obscene										3691
subsampling	0.64	0.74	0.68	0.61	0.75	0.68	0.46	0.53	0.49	
data augmentation	0.63	0.77	0.69	0.60	0.78	0.68	0.52	0.50	0.51	
threat										211
subsampling	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
data augmentation	0.27	0.67	0.38	0.22	0.66	0.34	0.11	0.00	0.01	
insult										3427
subsampling	0.59	0.60	0.60	0.57	0.60	0.59	0.45	0.48	0.47	
data augmentation	0.55	0.68	0.61	0.52	0.67	0.58	0.48	0.46	0.47	
identity_hate										712
subsampling	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
data augmentation	0.59	0.24	0.34	0.45	0.11	0.17	0.48	0.02	0.04	

4.2 Risultati Classificazione binaria della tossicità e classificazione multi-label del tipo di tossicità

Per questo approccio, essendo di classificazione binaria è stata utilizzata l'*accuracy*, come mostrato in Figura 5.

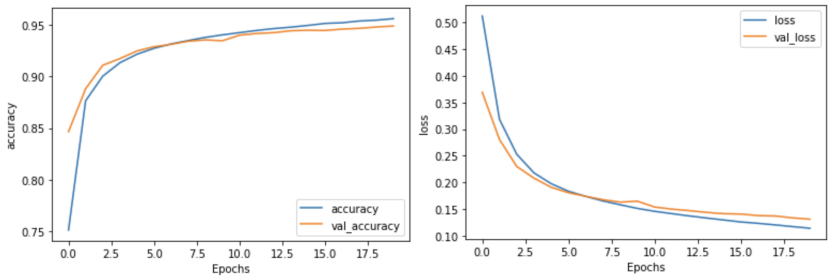

Figura 5: Accuracy e loss in caso di modello binario.

Come fatto in precedenza si confrontano i risultati della predizione con i dati del test-set. Le metriche utilizzate sono precision, recall e F1-score, mentre viene abbandonata l'accuracy in quanto il test-set rimane sbilanciato.

Tabella 8: Precision, recall e F1 score in caso di modello binario.

Label	precision	recall	F1 score	support
non toxic (0)	0.99	0.87	0.93	57888
toxic (1)	0.43	0.92	0.59	6090

La fase successiva di classificazione multi-label per la tipologia di tossicità (Figura 6) è stata realizzata con le stesse metriche utilizzate nel primo approccio.

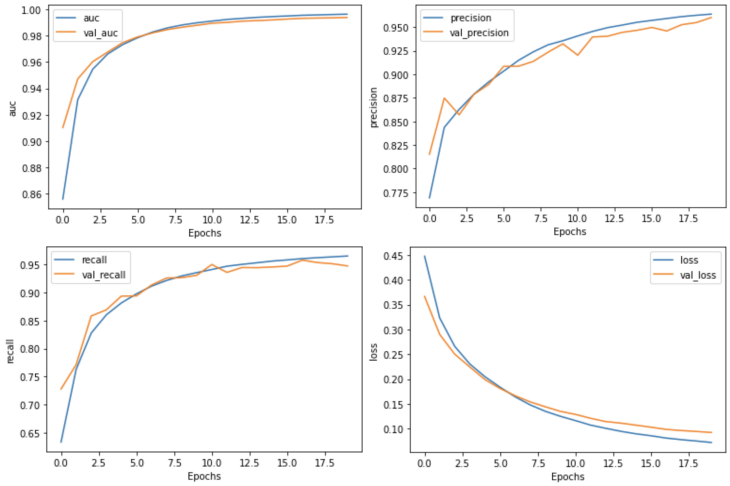

Figura 6: AUC, precision, recall e loss in caso di modello multi-label per determinare la sottocategoria.

Si è verificata quindi la bontà del modello confrontando predizioni e valori reali per le cinque sottocategorie, ottenendo i risultati mostrati in Tabella 9.

Tabella 9: Valutazione del modello tramite test set su un totale di 6090 commenti.

Label	precision	recall	F1 score	support
severe_toxic	0.23	0.44	0.30	367
obscene	0.78	0.76	0.77	3626
threat	0.46	0.48	0.47	205
insult	0.70	0.67	0.68	3342
identity_hate	0.60	0.51	0.55	693

5 Discussione

Confrontando i valori delle metriche ottenute dai tre modelli in Tabella 7, si nota che il dataset ottenuto con la tecnica di data augmentation offre prestazioni nettamente migliori di quelle ottenute con subsampling. Osservando la medesima tabella si nota come le RNN ottengano risultati migliori in termini di F1-score rispetto al modello CNN. L'F1-score è maggiore nelle architetture GRU rispetto alle LSTM per la classi *toxic* e *severe toxic* ma senza distaccarsi troppo. Viceversa il modello LSTM presenta valori leggermente migliori per le classi *obscene*, *threat*, *insult* e *identity hate*. Il medesimo comportamento si può riscontrare anche per precision e recall. Per il modello binario, analizzando la Figura 5 si può notare che l'inserimento del dropout permette di evitare l'overfitting sui dati. Valutando le perfomance sul test set, visibili in Tabella 8, si può notare che i risultati sono simili a quelli valutati nell'approccio precedente. Per quanto riguarda la classificazione delle sottocategorie di tossicità si sono osservati lievi miglioramenti per tutte e tre le metriche. In generale applicando i modelli sul test set per valutarne il potere predittivo si può notare come visibile nelle Tabelle 7, 8 e 9 che le metriche risultano peggiori rispetto a quelle ottenute sul train e sul validation set poiché il test set è fortemente sbilanciato.

6 Conclusioni

L'obbiettivo primario del modello realizzato è quello di disincentivare, o nel caso necessario di bloccare, la pubblicazione di commenti tossici. Per questo motivo la soluzione ideale da implementare in un sito o in una piattaforma social dovrebbe essere in grado di riconoscere in anticipo la tossicità dei commenti.
Il modello che risulta migliore per quanto mostrato fino a questo momento è l'LSTM applicato al secondo approccio. Questa architettura è da considerarsi migliore nonostante non ci siano grosse differenze tra i due approcci nell'individuazione della tossicità di un commento. A essere discriminante è la capacità del modello di individuare, con un F1-score maggiore, le tipologie di tossicità tenendo conto della loro rarità.
L'F1-score, infatti, è la misura più utile da tenere in considerazione in caso di problemi con classi sbilanciate essendo una media armonica di precision e recall. Oltre all'F1-score è considerato particolarmente indicativo anche il valore della recall: questa, infatti, essendo calcolata come rapporto tra i veri positivi e gli appartenenti alla classe positiva, tiene conto dell'efficacia nell'associare correttamente la predizione di un elemento alla classe a cui appartiene realmente. Ciò è ritenuto fondamentale nel problema analizzato in quanto è preferibile segnalare come potenzialmente tossico un commento che in realtà non lo è, piuttosto che non accorgersi della pubblicazione di un commento tossico.
Un eventuale sviluppo della soluzione proposta potrebbe riguardare un approfondimento legato al grado di tossicità dei commenti. Sarebbe interessante, infatti, poter stabilire non solo la probabilità che un commento sia tossico, ma anche di quanto lo sia. L'introduzione di questo controllo permetterebbe quindi a un algoritmo preposto alla moderazione di siti e social di modulare risposte differenti. Sarebbe possibile, ad esempio, suggerire nella fase precedente alla pubblicazione di riformulare il commento in modo più conciliante se questo fosse solo parzialmente tossico. Nel caso in cui un commento contenesse invece minacce o pesanti insulti rivolti all'identità della persona si potrebbe addirittura impedire direttamente che questo venga pubblicato.

Riferimenti bibliografici

[1] Spiros V. Georgakopoulos et al. "Convolutional Neural Networks for Toxic Comment Classification". In: *CoRR* abs/1802.09957 (2018). arXiv: 1802.09957. URL: http://arxiv.org/abs/1802.09957.

[2] Alon Jacovi, Oren Sar Shalom e Yoav Goldberg. "Understanding Convolutional Neural Networks for Text Classification". In: *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Brussels, Belgium: Association for Computational Linguistics, nov. 2018, pp. 56–65. DOI: 10.18653/v1/W18-5408. URL: https://www.aclweb.org/anthology/W18-5408.

[3] Marwan Torki, Mai Ibrahim e Nagwa El-Makky. "Imbalanced Toxic Comments Classification Using Data Augmentation and Deep Learning". In: dic. 2018. DOI: 10.1109/ICMLA.2018.00141.