Detecció de depressió a xarxes socials mitjançant varis mètodes de Machine Learning

Martí Caixal i Joaniquet

1 INTRODUCCIÓ

Les xarxes socials són considerades com uns sistemes d'informació en línia que permeten compartir l'estil de vida dels seus usuaris. Cada un té el seu perfil personal on penja actualitzacions del seu dia a dia i la resta d'usuaris poden reaccionar-hi i posar comentaris. De la mateixa manera, també es permet seguir a gent i altres institucions que siguin de l'agrat d'un [1].

L'inici de les xarxes socials es remunta a finals dels anys 90, quan l'internat tot just passava de ser una eina orientada a professionals a ser d'àmbit general. Tot i no ser la primera en aparèixer, la xarxa "MyS-pace" va ser la que va popularitzar aquest fenomen i va obrir pas a un seguit de noves xarxes socials. La més famosa, i que actualment segueix sent la que té més usuaris actius, és "Facebook", creada per en Mark Zuckerberg. Si bé inicialment l'únic objectiu era estar en contacte amb la gent del teu cercle més proper, avui en dia les xarxes socials són un mitjà per la gent famosa on rebre milers i milers de seguidors i visualitzacions, deixant de banda l'objectiu principal amb el que es van crear. [2]

Les xarxes socials no estan exemptes de problemàtiques. Al cap i a la fi, són un lloc on tothom pot dir la seva sense cap tipus de restricció. Això ha portat fins a un punt on la gent diu allà el que no és capaç o bé no s'atreveix a dir en persona.

Sí bé aquestes xarxes tenen codis de conducte i contenen d'equips de moderadors, no es pot fer front a tots els problemes. Al fet que se li dona més importància i es destinen més recursos de forma activa és a l'anomenat "cyber-bullying", doncs és el que més canta i no deixa de ser un atac des d'un individu cap a un altre. Si més no, també hi ha altres problemes que potser no són tant cridaners, però estan en molta més quantitat. Un d'ells és la depressió que pateixen molts dels seus usuaris. Com bé ja s'ha comentat, les xarxes socials permeten posar comentaris a internet, sent un lloc perfecte per la gent amb problemes d'ànims o de depressió per poder expressar-se i deixar anar tot els que els hi preocupa. [3]

Un estudi demostra que més d'un 20% dels usuaris han penjat comentaris amb indicis que podrien estar patint depressió o similar. No només això, sinó que és una moda que està en augment, havent-hi el doble de casos, proporcionalment, ara que fa 10 anys. Addicionalment, hi ha hagut alguns casos on els usuaris expliquen les "penúries" del seu dia a dia fins al punt on escriuen allà la mateix la nota de suïcidi. [4]

Clarament, totes aquestes notícies han provocat un seguit de queixes a les empreses propietàries de les xarxes per part de moltes organitzacions i institucions. Els responsables de moderació de les xarxes socials es defensen dient que no hi ha manera de poder veure tots els posts amb indicis de depressió. A diferència dels que contenen "cyber-bullying" o similars, que són reportats per altres usuaris (normalment

les víctimes), els missatges amb continguts depriments passen desapercebuts, o simplement no se'ls hi dona importància, per la resta d'usuaris.

Aquest fet dificulta moltíssim la feina dels equips moderadors, els quals no tenen els mitjans necessaris per avaluar tots els missatges i comentaris. No només això, sinó que, al no estar incomplint cap normativa, tampoc poden prendre cap acció al respecte.

2 OBJECTIUS

Així doncs, hi ha la necessitat d'obtenir un sistema que pugui fer front al problema esmentat anteriorment. Per la seva pròpia naturalesa, s'ha de solucionar no pas actuant un cop passa, sinó de forma preventiva prenent accions abans de que sigui massa tard.

L'objectiu és poder identificar els casos d'usuaris que necessitin ajuda mitjançant models predictius basats en intel·ligència artificial. Més específicament, cal treballar i investigar l'anomenat "Natural Language Processing", traduït a processament de llenguatge natural.

Clarament, ja hi ha molts mètodes i models disponibles que realitzen la tasca desitjada. L'objectiu, per tant, no és crear des de zero un nou mètode, sinó fer un estudi de l'eficiència i èxit que tenen cadascun d'ells. Per tant, s'implementaran un seguit de mètodes diferents i es procedirà a fer les proves adients. Les dades utilitzades són datasets ja classificats correctament. Aquestes dades són extretes directament i sense tractar de les xarxes socials Twitter i Reddit. Aquest fet per una banda permet tenir una representació pràcticament exacte de les dades amb les que s'enfronten els varis models en el moment de la veritat. Per una altra banda, al ser informació sense tractar, també obre la porta a fer un Exploratory Data Analysis (EDA) i treure ja unes estadístiques i característiques preliminars, les quals després es podran comparar amb els resultats arribats un cop executats els models. Addicionalment, els models es posaran a prova tant amb les dades sense tractar, com fent un previ tractament del dataset amb l'objectiu de veure el nou comportament dels models i si hi ha algun indici de millora a les produccions.

Per poder arribar a tots els objectius detallats a continuació, caldrà fer ús d'un conjunt de dades que tinguin la informació amb la que després s'hauran d'enfrontar els models. Des del portal "Kaggle", on es poden trobar molts datasets de qualsevol àmbit, se'n descarreguen uns quants:

- Mental Hearlth Twitter [8]
- Depression social media [9]
- Depression: Reddit Dataset [10]
- Sentimental Analysis for Tweets [11]

Se'n agafa més d'un perquè tots tenen alguna característica especial que els fa únics. El primer és el que es podria considerar més interessant, doncs, a part del text del tweet, té un seguit d'atributs com el número de seguidors o número de tweets diaris. D'aquesta manera permetrà veure realment fins a quin punt el NLP és efectiu per trobar cassos de depressió a xarxes socials. Primer se li s'aplicarà únicament el sentiment analysis, i després també s'utilitzaran la resta d'atributs per veure si hi ha gaire millora. El segon

es caracteritza per estar classificats en una escala de valors que defineixen el nivell de depressió, a diferència de la resta on únicament tenen valors de “depressió” o “no depressió”. El tercer es caracteritza per estar ja netejat d’emojis, links d’internet i similars. Finalment, el quart únicament conté el tweet i la classificació.

Dins del Machine Learning ja s’ha fet un seguit d’investigacions i treballs des dels quals es basaran els objectius.

Per una banda hi ha els mètodes tradicionals de Machine Learning. L’article “Sentiment Analysis of Review Datasets using Naïve Bayes’ and K-NN Classifier” [5] explora tant Naïve Bayes i K-NN, intentant evaluar el rendiment i resultats de cada un. Tant un com l’altre es comporten similarment i es postulen com a bones opcions dins d’aquest tipus de mètodes, però sent el Naïve Bayes el que ho fa millor amb el 90% d’accuracy. Tot i això, l’estudi d’aquests mètodes s’orienta a fer un sentiment analysis d’opinions de pel·lícules, no pas de posts a les xarxes socials. També hi ha més estudis, com el “Sentiment Analysis on Twitter Data using KNN and SVM" [6] que sí que fan el sentiment analysis sobre un cas de xarxes socials. Per aquest últim, el mètode de SVM és el que dona més bons resultats. Tot i això, no s’està buscant específicament depressió, sinó si el tweet és positiu o negatiu.

Un dels objectius als que es vol arribar és comparar varis models tradicionals i posar-los a prova directament en detectar depressió als posts en xarxes socials. D’aquesta manera es podrà veure si els que tenen més èxit en sentiment analysis genèric, també el tenen quan es busca un sentiment en concret. Més específicament, per aquesta part es posaran a prova els classificadors Naïve Bayes, Decision Tree, Random Forest, SVM i KNN.

Addicionalment, també hi ha maneres de millorar les prediccions de varis classificadors, l’article “Comparative Analysis of Bagging and Boosting Algorithms for Sentiment Analysis” [7] n’explora alguns. Els resultats són millors que aplicant un simple classificador, però, de nou, no posa pas èmfasi en la depressió. Per tant, en aquest apartat també es posarà a prova els mètodes d’ensamblament Boosting i Bagging aplicant-los als classificadors prèviament mencionats, amb l’objectiu de veure fins a quin punt es pot arribar a millorar les prediccions de depressió per cada classificador.

Finalment, aquests mètodes més tradicionals tenen un seguit de paràmetres, anomenats Hyper Parameters, amb els quals configurar el seu comportament. Segons l’article “Hyperparameter Search in Machine Learning” [12], aconseguir trobar els valors adequats pot marcar la diferència entre unes bones prediccions i unes males. Caldrà veure fins a quin punt una bona elecció de Hyper Parameters pot afectar a les prediccions quan es busca un sentiment anàlisis de depressió.

Per una altra banda, també hi ha els mètodes basats en el Deep Learning que actualment estan triomfant més. L’estudi “Comparing Machine Learning and Deep Learning Approaches on NLP Tasks for the Italian Language” [13] parla sobre les diferències que hi ha entre el Deep Learning i els mètodes tradicionals de Machine Learning, concloent que per NLP el Deep Learning no acaba de tenir superioritat en les tasques de classificació que depenent molt en l’anàlisi semàntic.

Un mètode que ja porta un temps utilitzant-se per NLP són les anomenades Recurrent Neural networks (RNN). Es diferencien de les xarxes neuronals convencionals, i són especialment útils en NLP, pel fet que el input és una sola paraula, donant flexibilitat per treballar amb diferents llargades a les frases. Hi ha dos models que fins fa poc eren els més utilitzats, anomenats LSTM i GRU. L'article "LSTM and GRU neural network performance comparison study" [17] explora aquests dos mètodes, on conclou que el fet de retenir informació els hi permet posicionar-se al podi de les RNN. No obstant, està comparant amb tasques de molts camps, sense entrar en detall al NLP i encara menys al Sentiment Analysis. Un dels objectius serà aplicar ambdós mètodes per detectar depressió a les xarxes socials.

L'article "Attention is All you Need" [14], publicat per Google el 2017, presenta un nou model de xarxes neuronals especialment innovador a l'apartat de Natural Language Processing. S'anomena "Transformer" i es basa processar dades d'entrada seqüencials però, a diferència de les Xarxes Neuronals Recurrents, processen tota l'entrada alhora. L'article "Overview of the Transformer-based Models for NLP Tasks"[16] fa especialment atenció en el rendiment dels models Transformer amb els problemes NLP, concloent que efectivament han millorat molt les prediccions respecte a la resta de mètodes utilitzats fins al moment. Un objectiu serà aplicar varis models Transformers (GPT, BERT i RoBERTa) i veure com es comporten a l'hora de detectar casos de depressió a les xarxes socials, comparant-los tant entre ells com amb la resta de mètodes provats anteriorment.

3 METODOLOGIA

Per poder desenvolupar el projecte de manera adequada i ordenada, és necessari establir una metodologia de treball la qual seguir, permetent així tenir un bon control del flux de treball i compliment de les tasques i entregues. La metodologia escollida és l'anomenada "àgil".

Es basa en iteracions curtes, adaptabilitat als canvis, facilitat de detectar i solucionar errors, i entregues parcials. És un projecte que es pot subdividir en diferents apartats independents entre ells i aquesta metodologia és especialment útil en aquests casos. D'aquesta manera, si una part presenta complicacions, la resta no es veuran pràcticament afectades. També permet fer, al final de cada iteració, un anàlisi de la situació actual, valorant el que s'ha realitzat, el que manca per realitzar i qualsevol aspecte que calgui modificar/adaptar.

Finalment, també es fa us d'un controlador de versions GIT, tenint així un registre i historial de tots els canvis i permetent revertir-los en cas de fer falta.

4. INFORME PROGRÉS 1

Preprocessament

Prèviament a començar a entrenar els models cal netejar i fer un preprocessament de les dades. Normalment les dades sense tractar són molt difícils de treballar i contenen informació irrellevant. Pel cas de sentiment analysis es realitza les següents modificacions:

- Eliminar noms d'usuari: Al estar tractant amb datasets que provenen de xarxes socials, moltes missatges contenen noms d'usuari, identificats pel símbol @ al començament de la paraula. Tots aquests noms d'usuaris cal que s'esborrin.
- Eliminar Stop Words: Són articles, adverbis i similars que no aporten significat, sinó que simplement ajunten paraules que si aporten informació com els adjectius o els noms.
- Eliminar números: els números que poden anar apareixent als missatges no aporten informació i compliquen les prediccions.
- Lemmanization: és un procés que busca donar un sol valor a totes les paraules que signifiquen el mateix. Un clar exemple són les diferents conjugacions d'un verb. Totes les conjugacions d'un mateix verb són passades al seu infinitiu. D'aquesta manera s'aconsegueix mantenir el significat i a l'hora simplificar les dades amb les que hauran de tractar els models.
- Puntuació: Els símbols de puntuació, com les comes o els punts, també són eliminats.

Feature extraction

Fins ara només s'ha modificat el contingut del missatge, però segueix tenint la mateixa forma (un seguit de paraules). Queda pendent extreure les característiques dels missatges per tenir un format que els models puguin entendre. Hi ha dos tècniques molt utilitzades: Bag-of-Words i TF-IDF [18].

El Bag-of-Words funciona creant una llista de totes les paraules que existeixen als missatges. Per cada missatge, indica el número de vegades que apareix cada paraula de la llista. El següent exemple mostra com funciona:

- Frase1: El cotxe que vaig comprar és gran i vermell.
- Frase2: Vaig comprar un ordinador molt gran i car.

	Cotxe	Comprar	Gran	Vermell	Ordinador	car
Frase1	1	1	1	1	0	0
Frase2	0	1	1	0	1	1

La taula resultant conté la freqüència de cada paraula a cada una de les frases. Cal destacar que aquest sistema té en compte només els número de vegades que apareix una paraula, però no la seqüència ni l'ordre.

L'altre mètode és el TF-IDF, que significa Term Frequency-Inverse Document Frequency. Funciona donant un pes (importància) a cada paraula.

Funciona en dos parts. La primera és Term Frequency, que és el número de vegades que una paraula apareix al text, dividit entre total de paraules del text. Si el text té 100 paraules i la paraula en qüestió apareix 3 vegades, el TF és 0.03.

$$TF_{x,y} = \frac{N_x}{N_y}$$

$$x = paraula$$

$$y = text$$

La segona part és Inverse Document Frequency. Es basa en calcular el logaritme del número de missatges al dataset entre el número de missatges on la paraula en qüestió apareix. Si el dataset té 100 missatges, i la paraula en qüestió apareix a 3 misstges, el IDF és 1.5

$$IDF_x = \frac{\log(N)}{N_x}$$

$$N = número\ de\ documents\ al\ dataset$$

$$N_x = número\ de\ documents\ que\ contenen\ la\ paraula\ X$$

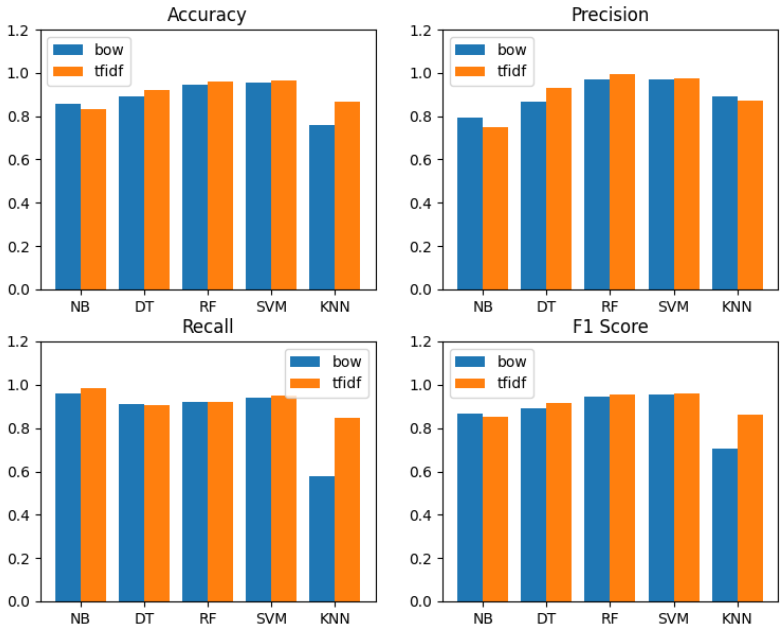
El valor final és TFIDF(paraula) = TF(paraula)*IDF(praula)

$$TF - IDF_w = TF_{x,y} * IDF_x$$

Primeres proves i mètriques a utilitzar

Les primeres proves realitzades són executant individualment els models comentats a la planificació: Naive Bayes, Decision Tree, Random Forest, SVM i k-nearest neighbours. A continuació es mostra els resultats obtinguts.

El primer dataset que es prova és el “Reddit”. La següent imatge presenta una visualització dels resultats. Tot i ser les primeres prediccions i no s’ha donat importància als paràmetres, ja es pot extreure algunes conclusions. Es pot veure que el model KNN té un rendiment inferior, mentre que la resta estan molt empatats.



Addicionalment, es pot aprofitar aquesta gràfica per parlar de les mètriques importants. Ara mateix se'n mostren 4:

- Accuracy: De totes les prediccions fetes, quantes són correctes?
- Precision: De les prediccions positives, quantes realment són positives?
- Recall: De totes les dades positives, quantes realment han estat classificades com a positives
- Valor únic per representar tant "Precision" com "Recall". Es calcula mitjançant una mitjana ponderada.

L'objectiu del projecte és poder detectar i analitzar els casos de depressió en xarxes socials. Per tant, és molt més important poder identificar tots els casos reals de depressió. La mètrica "recall" és la que permet saber si l'objectiu s'està complint i, per tant, és la que se li donarà més importància durant el projecte. És cert que no s'estarà tenint molt en compte els casos reals de no depressió classificats com a positius, però al cap i a la fi això són mètodes per passar un filtre inicial, i després cada cas és tractat personalment per persones qualificades (serveis socials, psicòlegs, ...). És per aquesta raó que és més important fer un filtre inicial que no deixi de banda a cap persona que realment necessiti ajuda, deixant per més endavant separar els classificats positius incorrectament.

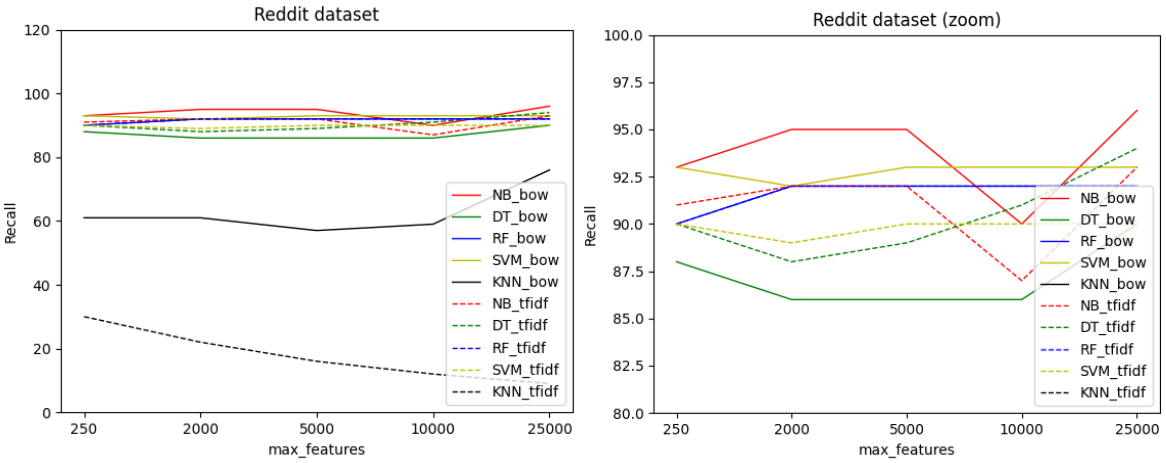
Finalment, cal comentar que el dataset "Twitter_13" no es pot utilitzar perquè està mal classificat.

Mètodes d'extracció de característiques i com afecten

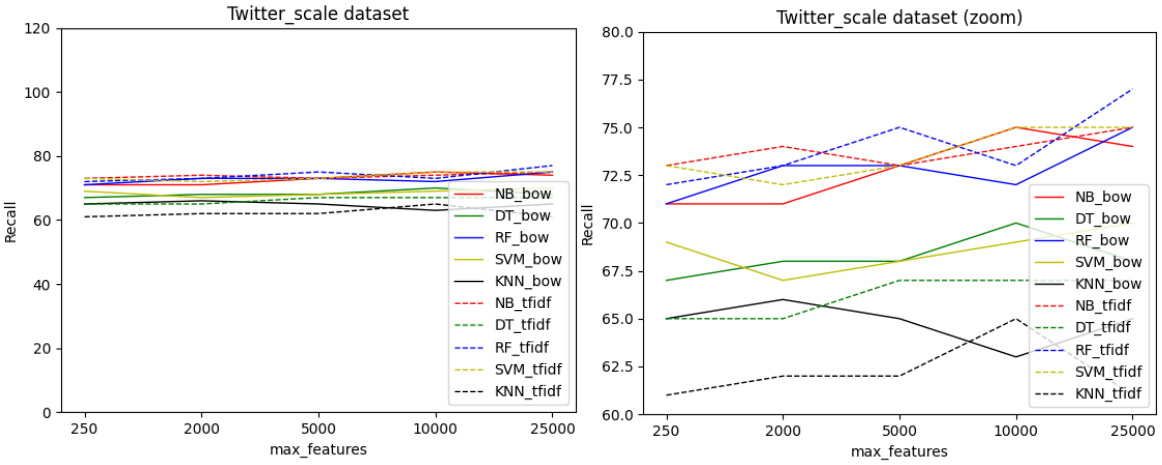
Un altre punt a comentar és el paràmetre "max_features" tant del "Bag of Words" com del "TF-IDF". Especifica el màxim número de paraules a extreure'n informació del text. El funcionament normal dels dos mètodes és utilitzar totes les paraules. Clarament, hi ha paraules que apareixen més vegades i aporten més informació que d'altres. Especificant un número es permet indicar el màxim número de paraules a agafar, sent sempre les que més informació aporten.

Els resultats a continuació mostren la diferència de les prediccions utilitzant varis valors pel paràmetre “max_features”.

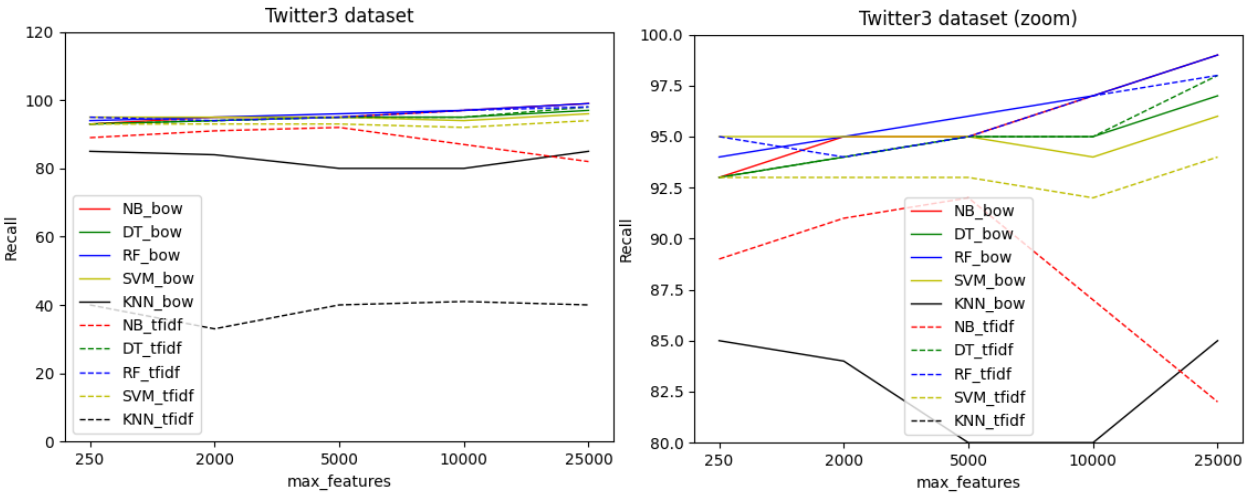
El dataset *Reddit* té els següents resultats. La primera imatge mostra tots els valors de la mètrica *recall*. La segona, en canvi, amplia la part superior per poder veure més bé el detall.



El dataset *twitter_scale* obté els següents resultats:



Finalment, el dataset *twitter3* obté els següents resultats:

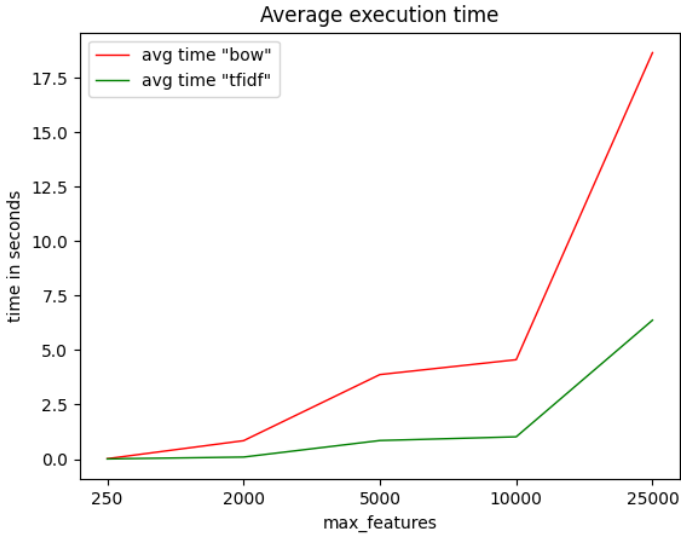


Veient els resultats dels tres datasets, el tipus de model utilitzat és molt més significatiu que no pas el mètode per extreure característiques. En canvi, el valor de *max_features* sí que pot afectar de forma molt significativa al temps d'execució.

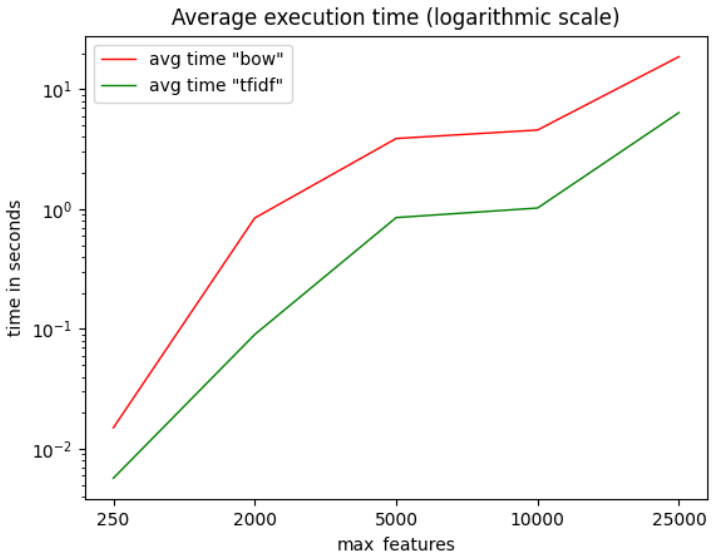
Per poder fer un seguiment de totes les proves i resultats que s'obtenen durant el desenvolupament, un arxiu ".csv" guarda els models, configuració i valors de les mètriques obtingudes. El format és el següent:

model	params	dataset	type	average	f1	time	hyperpara	datetime
RF	{"rf_max_depth": 86, "rf_n_estimator	clean_twitter_scale	tfidf max_features=250	macro	0.6994	0.88199	True	1/11/2022 16:22
SVM	{"svc_kernel": "linear", "svc_gamma"	clean_twitter_scale	tfidf max_features=250	macro	0.6994	3.55995	True	1/11/2022 16:23
RF	{"rf_max_depth": 83, "rf_n_estimator	clean_twitter_scale	tfidf max_features=250	macro	0.6994	1.058	True	1/11/2022 16:22

Utilitzant les dades guardades al document, es pot veure com el paràmetre “max_features” afecta al temps d’execució:



Canviant l’escala de l’eix Y a logarítmica es permet veure més bé com afecta quan els valors són petits:



Per una banda es veu que el número de característiques a extreure afecta significativament al temps d’execució. Quan el valor és molt elevant s’utilitzen pràcticament totes les paraules i, depenent de la quantitat de text a analitzar, pot arribar a trigar molts minuts. Utilitzant poques paraules els temps milloren molt, fins al punt que són mil·lèsimes de segon. No obstant, ja s’ha vist que els resultats obtinguts són pràcticament idèntics. Per altra banda, el mètode “TF-IDF” permet tenir resultats entre 2 i 3 vegades més ràpid que el “Bag of Words”.

El primer gràfic de temps d’execució pot fer pensar que no hi ha gaire diferència de temps entre 250 i 2000 pel valor de *max_features,* portant a utilitzar el segon valor ja que dona uns resultats un pel millors. Realment sí que hi ha bastanta diferència entre ambdós valors, podent-se comprovar al segon

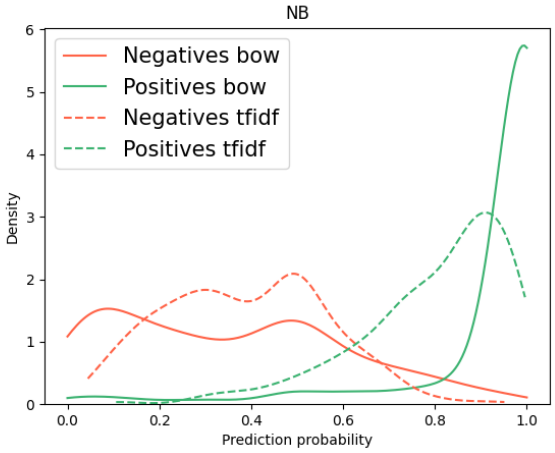
gràfic, que fa servir una escala logarítmica, que hi ha una diferència de pràcticament 10 vegades més temps entre un i l'altre.

Confiança de les prediccions

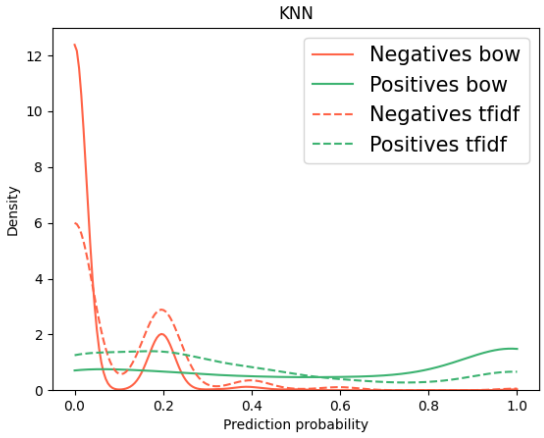
Veient els resultats de les prediccions i els diferents datasets, sembla que el problema ja està resolt en més d'un cas. Les mètriques obtingudes són sorprenentment bones per dos dels tres datasets utilitzats.. Tot i això, cal veure amb quina confiança dona aquests resultats. Al cap i a la fi, s'està buscant la probabilitat de que una mostra pertanyi a cada classe, sent classificada a la classe que tingui la probabilitat més alta.

Ara sí cal analitzar-ho model per model, doncs els resultats són bastant diferents. També es compara utilitzant tant "BoW" com "TF-IDF". A continuació es mostren els resultats de les prediccions del dataset "reddit".

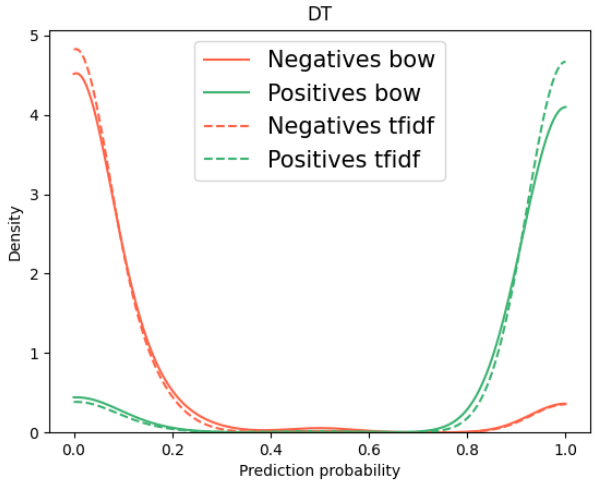
El "Naive Bayes", mostrat a continuació, demostra que realment les prediccions no són gaire confiables. Analitzant primer les prediccions positives, el mètode *bow* funciona molt més bé que el *tfidf*, doncs té la gran part de les prediccions amb una probabilitat superior al 80%. De totes formes, les mostres positives tenen en general millors probabilitats que les negatives. Tant per *bow* com per *tfidf* la probabilitat de les mostres negatives està distribuïda de forma igual entre el 0% i el 50%, tenint inclús unes quantes mal classificades per sobre del 50%.



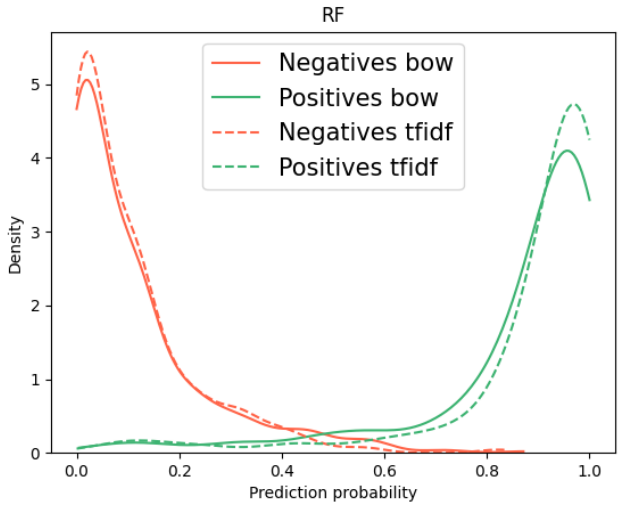
El "KNN", com era d'esperar veient les anteriors gràfiques, té una confiança molt baixa en les prediccions. Aquí es pot veure que el problema el té en els True Positive, doncs les prediccions negatives les fa bé. En canvi, les prediccions positives tenen la probabilitat distribuïda de forma pràcticament igual entre el 0% i el 100%.



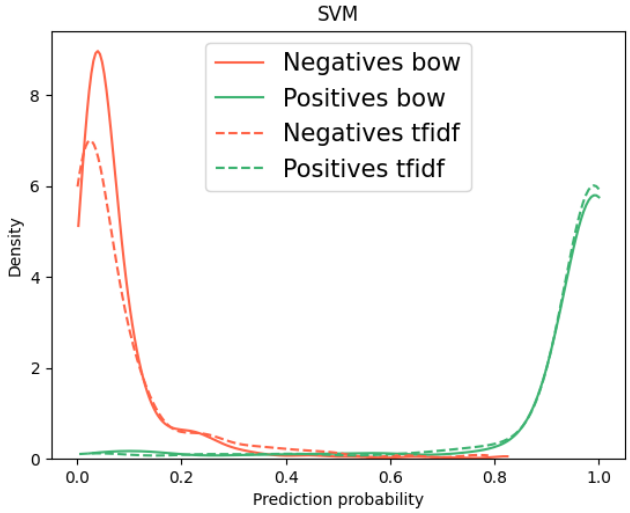
El model “Decision Tree” té els resultats més confiables fins al moment. Tant per les mostres positives com per les negatives, pràcticament la totalitat d’elles estan correctament classificades amb una probabilitat superior al 85%. És interessant comentar que no hi ha mostres classificades prop del 50% de probabilitat, però sí que hi ha algunes mal classificades a l’altre extrem d’on haurien d’estar.



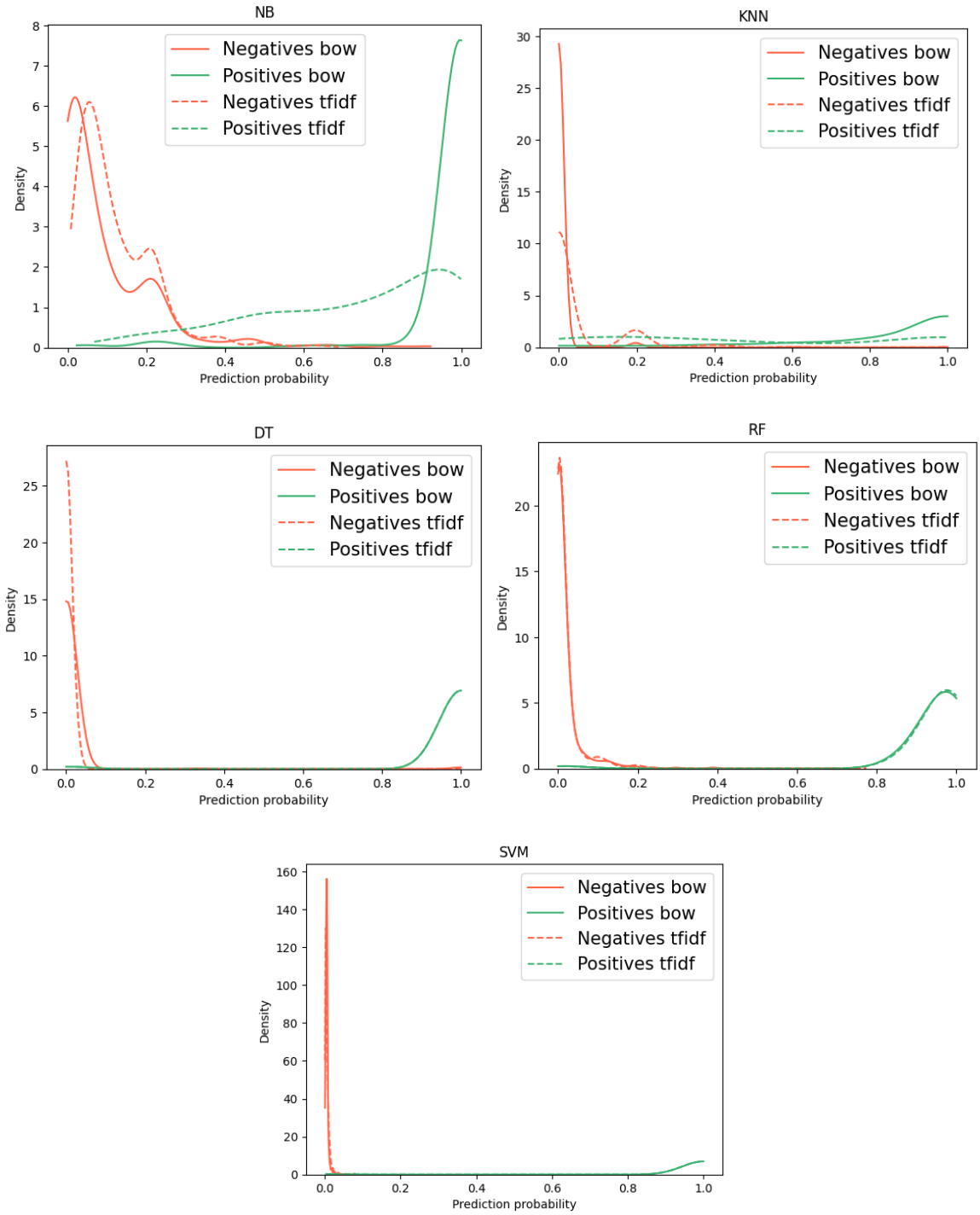
El “Random Forest” és molt similar al Decision Tree. Al tractar-se d’un conjunt de Decision Trees, la confiança es queda més centrada ja que es fa una votació d’un seguit de votacions prèvies. Ara ja no es té tantes mostres amb una confiança propera al 100%, però també s’elimina el problema on hi havia mostres mal classificades a l’extrem oposat.



Finalment, els models SVM presenten una bona confiança pels dos casos. Si bé no hi ha tantes mostres classificades prop del 100% com altres models, segueixen estant a prop i alhora es lliure de tenir mostres mal classificades a l'altre extrem que no els hi pertoca.

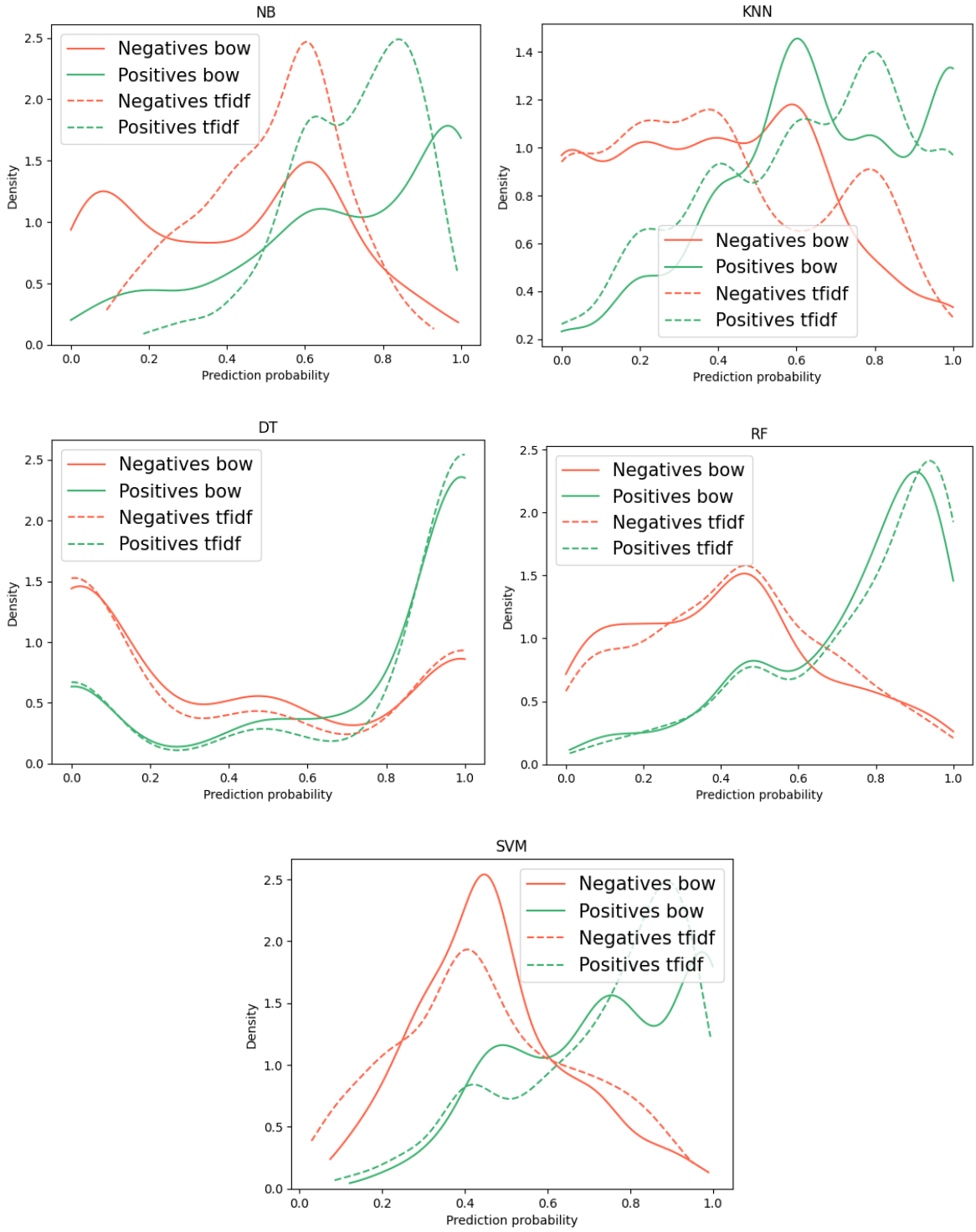


Pel dataset "Twitter3" el comportament és molt similar. Al tractar-se d'un dataset que prèviament ja s'ha vist que obté més bones prediccions, alguns problemes desapareixen. Per exemple, el model *Decision Tree* ja no presenta una part de les mostres a l'altre extrem de probabilitat on li pertocaria ser. Addicionalment, ja es pot començar a veure com la utilització de *bow* o *tfidf* és important depenent del model a utilitzar. Els dos datasets tenen formats diferents, un sent missatges de twitter i l'altre publicacions de Reddit. Tot i això, els models funcionen en els dos casos millor en un mètode que l'altre. És d'especial atenció el model *Naive Bayes* que quan utilitzar *tfidf* té problemes per realment classificar les mostres positives amb confiança.



El dataset ´Twitter_scale" és el que pitjor resultats ha donat abans i aquí es veu reflectit que la confiança en que ha fet les prediccions és molt baixa.

De totes formes, un cop havent fet la prova amb 3 datasets diferents, es veu que el mètode *tfidf* dona més bones prediccions amb millor confiança, alhora que redueix el temps d'execució a una tercera part del que triga el *bow*. L'únic model que té molta diferència entre els dos mètodes d'extracció de característiques és el *Naive Bayes*, on *tfidf* dona molta incertesa a les prediccions de classe positiva.

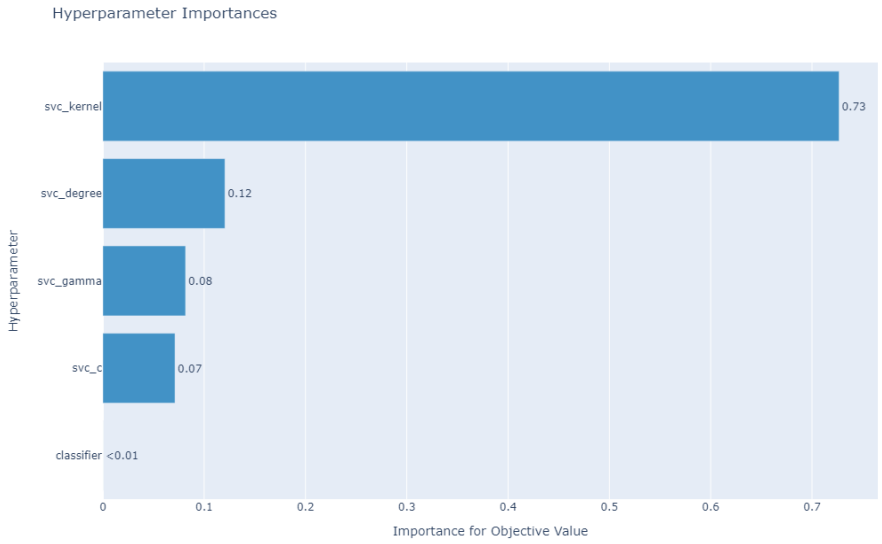

NB
KNN
DT
RF
SVM
Negatives bow
Positives bow
Negatives tfidf
Positives tfidf
Density
Prediction probability

Hyperparameter Tuning

Per realitzar la part de Hyper-Parameter Tuning es fa servir el framework Optuna [19]. Resumidament, és una eina per Python que permet fer una cerca òptima dels Hyper-parametersk. En comptes de provar valors aleatoris, fa una aproximació basant-se en proves anteriors i poda tots els camins que no són prometedors. Això, juntament amb un ús eficient de paral·lelització, dona uns resultats molt ràpids i òptims.

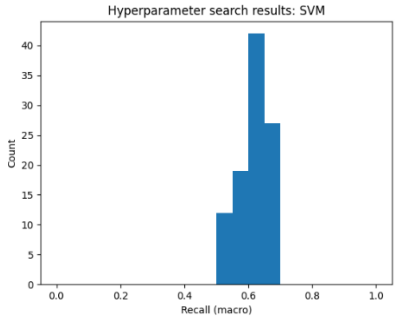
Els resultats obtinguts, però, no són gaire útils, les mètriques obtenen pràcticament els mateixos valors. De totes maneres, ara es pot visualitzar quins són els paràmetres que més afecten a les prediccions de cada model.

El SVM, per exemple, té el comportament molt afectat pel kernel utilitzat. En les proves realitzades el kernel "rbf" és el que ha donat les millors prediccions.

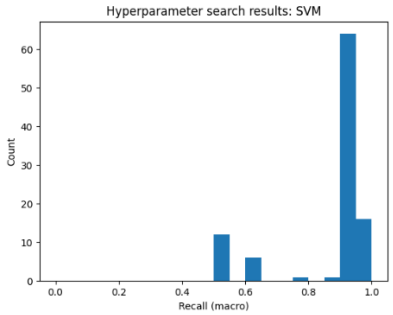


{"svc_kernel": "rbf", "svc_gamma": 0.73402651786784, "svc_c": 6.409289761666191, "svc_degree": 1}
{"svc_kernel": "rbf", "svc_gamma": 0.47705514303358026, "svc_c": 5.252992686178238, "svc_degree": 1}
{"svc_kernel": "rbf", "svc_gamma": 0.8041805255834942, "svc_c": 4.775910070060698, "svc_degree": 1}
{"svc_kernel": "rbf", "svc_gamma": 0.5434337543768762, "svc_c": 8.102470541180836, "svc_degree": 1}
{"svc_kernel": "rbf", "svc_gamma": 0.8003880906870993, "svc_c": 45.486036390042315, "svc_degree": 1}
{"svc_kernel": "rbf", "svc_gamma": 0.8039958212552204, "svc_c": 52.69563370435398, "svc_degree": 1}
{"svc_kernel": "rbf", "svc_gamma": 0.5232500515395961, "svc_c": 2.2291687754825067, "svc_degree": 1}

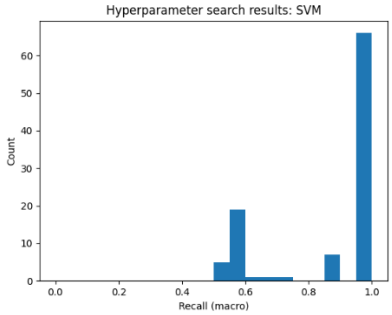
És també és l'únic model que té més varietat en les seves prediccions depenent dels paràmetres utilitzats. Com es veu a continuació, hi ha dos datasets que la majoria de vegades que s'ha entrenat i provat un model les mètriques són elevades, però hi ha unes quantes prop del 50%. Totes aquestes es donen quan el valor del paràmetre *c* és molt petit (inferior a 0.1)



Twitter_scale

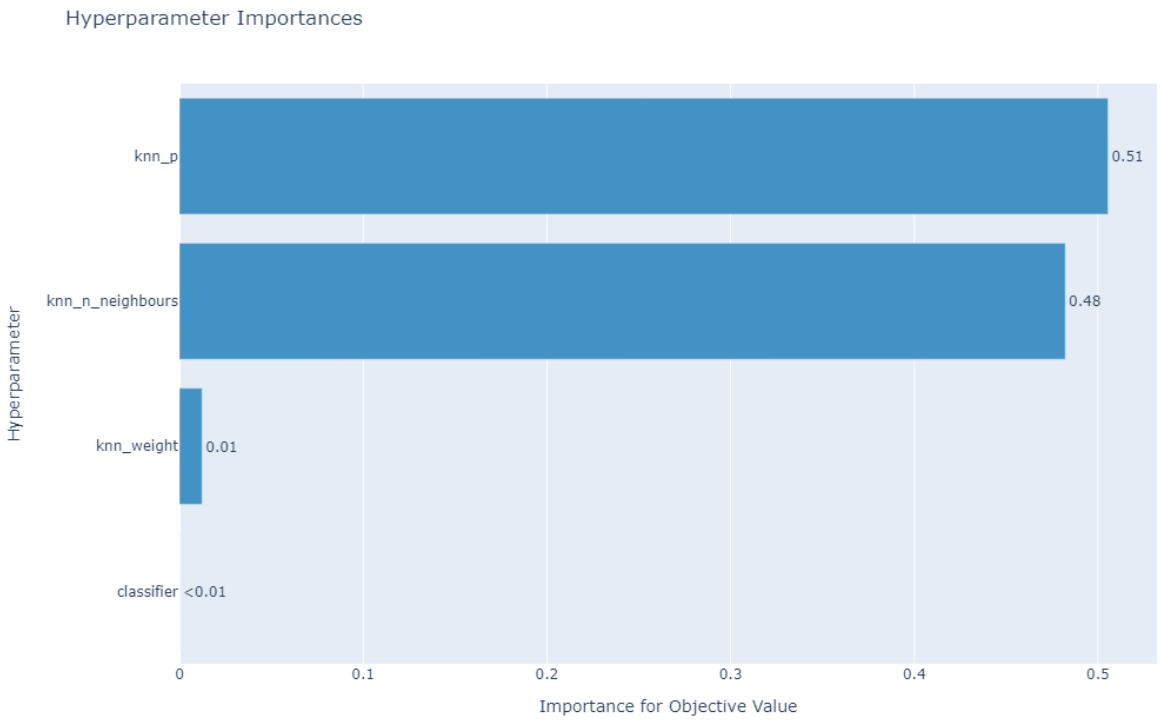


Reddit



Twitter_3

En canvi, hi ha alguns altres models, com el KNN, que les seves prediccions no es veuen tant afectades pels paràmetres utilitzats. Tot i no tenir un únic paràmetre que decideixi el comportament, les prediccions obtingudes estan totes reduïdes en un rang més petit.



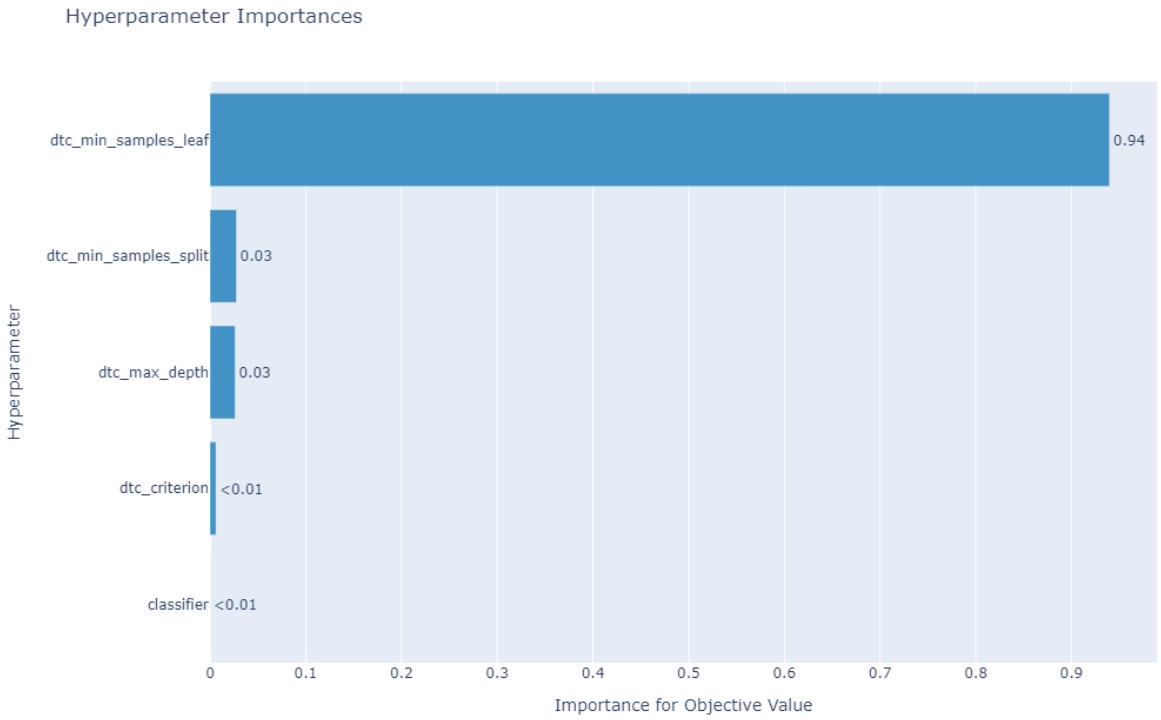
{"knn_n_neighbours": 3, "knn_p": 3, "knn_weight": "uniform"}
{"knn_n_neighbours": 3, "knn_p": 3, "knn_weight": "uniform"}
{"knn_n_neighbours": 4, "knn_p": 3, "knn_weight": "distance"}
{"knn_n_neighbours": 6, "knn_p": 3, "knn_weight": "distance"}
{"knn_n_neighbours": 8, "knn_p": 3, "knn_weight": "distance"}
{"knn_n_neighbours": 5, "knn_p": 3, "knn_weight": "distance"}
{"knn_n_neighbours": 5, "knn_p": 3, "knn_weight": "uniform"}

Tot i que les prediccions siguin sempre molt semblants, els millors resultats es donen en 2 situacions diferents:

a) Weight: 'uniform', p: 3 (Minkowski distance), n_neighbours: <=5
b) Weight: 'distance', p: 3 (Minkowski distance), n_neighbours >=9

En els casos en que s'utilitza un pes uniforme però amb gran quantitat de veïns, o a l'inrevés, els resultats empitjoren. Addicionalment, la distància de Minkowski és la única que es troba en el top 10% dels resultats del KNN.

Altres mètodes com el *Decision Tree* es veuen afectats pràcticament per un sol paràmetre:

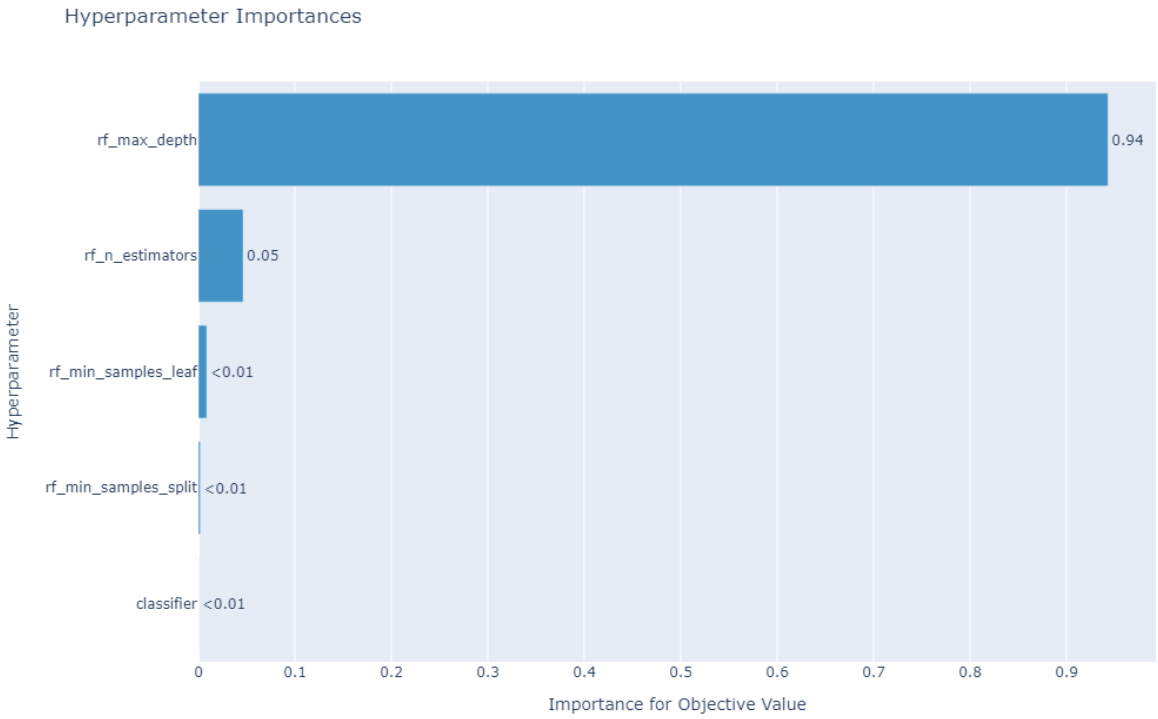


{"dtc_max_depth": 61, "dtc_min_samples_leaf": 9, "dtc_criterion": "entropy", "dtc_min_samples_split": 3}
{"dtc_max_depth": 64, "dtc_min_samples_leaf": 9, "dtc_criterion": "entropy", "dtc_min_samples_split": 3}
{"dtc_max_depth": 60, "dtc_min_samples_leaf": 9, "dtc_criterion": "entropy", "dtc_min_samples_split": 3}
{"dtc_max_depth": 45, "dtc_min_samples_leaf": 1, "dtc_criterion": "entropy", "dtc_min_samples_split": 3}
{"dtc_max_depth": 67, "dtc_min_samples_leaf": 7, "dtc_criterion": "entropy", "dtc_min_samples_split": 4}
{"dtc_max_depth": 47, "dtc_min_samples_leaf": 7, "dtc_criterion": "entropy", "dtc_min_samples_split": 4}
{"dtc_max_depth": 48, "dtc_min_samples_leaf": 9, "dtc_criterion": "entropy", "dtc_min_samples_split": 4}

En aquest cas és el "dtc_min_samples_leaf". Les proves realitzades que han donat les millors mètriques totes tenen pel paràmetre un valor inferior a 10, i van empitjorant a mesura que el paràmetre

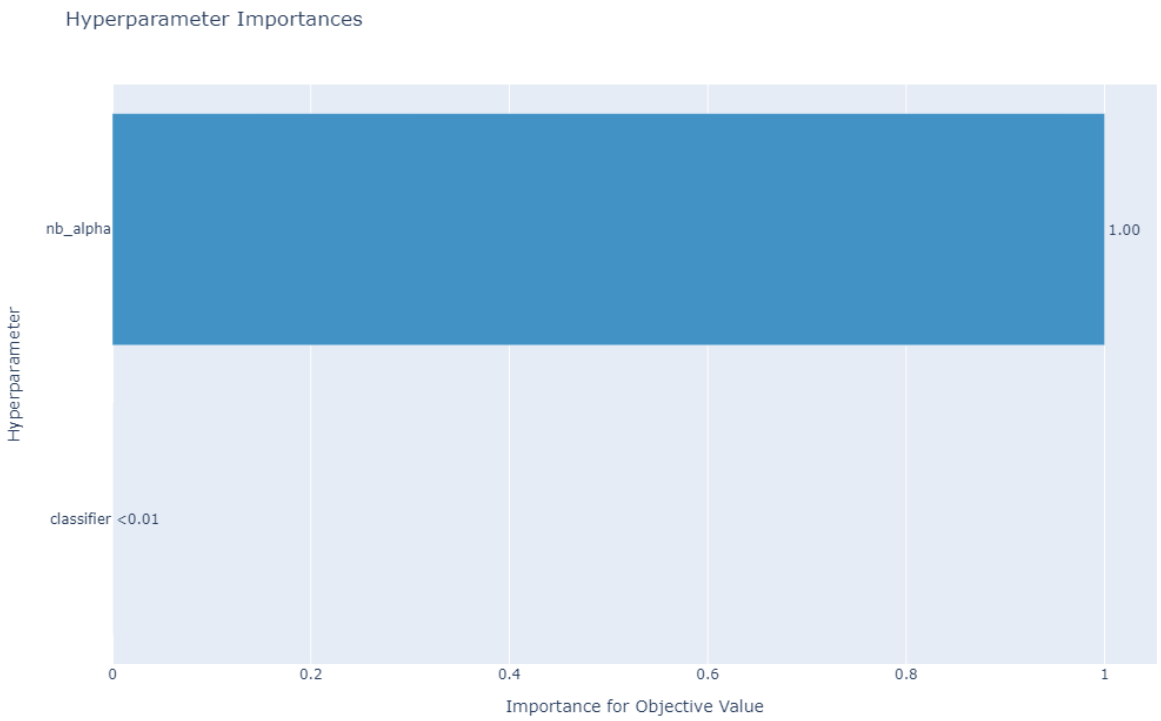
augmenta. També cal mencionar que la criteria utilitzada en totes les proves amb bon resultat és l'entropia, mentre que l'índex Gini es queda per sota respecte les mètriques.

El *Random Forest*, igual que passa amb el *Decision Tree*, es veu afectat principalment per un sol paràmetre



{"rf_max_depth": 47, "rf_n_estimators": 23, "rf_min_samples_leaf": 3, "rf_min_samples_split": 5}
{"rf_max_depth": 28, "rf_n_estimators": 30, "rf_min_samples_leaf": 1, "rf_min_samples_split": 7}
{"rf_max_depth": 32, "rf_n_estimators": 30, "rf_min_samples_leaf": 1, "rf_min_samples_split": 8}
{"rf_max_depth": 49, "rf_n_estimators": 19, "rf_min_samples_leaf": 2, "rf_min_samples_split": 5}
{"rf_max_depth": 28, "rf_n_estimators": 26, "rf_min_samples_leaf": 1, "rf_min_samples_split": 7}
{"rf_max_depth": 44, "rf_n_estimators": 24, "rf_min_samples_leaf": 2, "rf_min_samples_split": 6}
{"rf_max_depth": 44, "rf_n_estimators": 23, "rf_min_samples_leaf": 3, "rf_min_samples_split": 5}

Finalment el *Naive Bayes* té el paràmetre *alpha* per configurar el model.



{"nb_alpha": 0.0074749899724023}
{"nb_alpha": 0.0012161141396562502}
{"nb_alpha": 0.0018830314705746475}
{"nb_alpha": 0.005862946278312197}
{"nb_alpha": 0.005247004551393668}
{"nb_alpha": 0.00583714385514146}
{"nb_alpha": 0.005967275794414782}

Tot i aparèixer com a millors paràmetres sent un valor molt petit, no és fins que el paràmetre passa del valor 5 que els resultats realment empitjoren. Al cap i a la fi, utilitzar un valor molt alt comporta portar les probabilitats cap a 0.5 de cada classe i no interessa.

5 PLANIFICACIÓ

Iteració 1:

L'objectiu és començar amb el TFG, fer un estudi inicial de l'estat de l'art, definir els objectius, metodologia a utilitzar durant tot el desenvolupament, especificar les tasques de la resta d'iteracions.

SETMANA	TASCA	RESULTATS

Setembre, 3	• Reunió Inicial	• Fet
Setembre, 4	• Informe inicial (Objectiu) • instal·lar LaTex	• Informe inicial pràcticament finalitzat.

Iteració 2:

L'objectiu d'aquesta iteració és treballar en els mètodes tradicionals de Machine Learning i obtenir resultats de les prediccions de depressió a les xarxes socials. Per una banda s'implementaran els models definits a les tasques, i per una altra banda es mirarà de fer millores mitjançant tècniques com ensambles o hyper-paramter tuning.

Octubre, 1	• Primera sessió de seguiment • Buscar informació i estat de l'art • Definir objectius • Detallar planificació i tasques	• Informe inicial finalitzat i entregat. • Reunió Feta.

Octubre, 2	• Preparar entorn de desenvolupament a l'ordinador
	  ○ Instal·lar Python i IDE.
	  ○ Instalar llibreries necessaries per aquesta iteració.
	• Realitazar un Exploratory Data Analysis dels datasets amb els que es treballarà.
	• Aplicar els classificadors Naïve Bayes
	• Aplicar el classificador Decision Tree
	• Treballar en l'informe de progrés 1.
		• Python i llibreries instal·lats.
		• Els 3 classificadors s'apliquen i es fan prediccions.
		• Ha calgut dedicar molt temps (no previst) en fer un preprocessament de les dades i feature extraction.
		• No s'ha treballat en l'informe de progrés 1.

Octubre, 3	• Aplicar el classificador Random Forest • Aplicar el classificador Support Vector Machine • Aplicar el classificador K Nearest neighbour • Seguir treballant en l'informe de progrés 1.	• S'han aplicat els 2 classificadors restants i s'han fet les corresponents prediccions. • Ha fet falta millorar el preprocessament de les dades, tant per temps d'execució com per millorar els resultats. • Pràcticament no he treballat en l'informe de progrés 1
Octubre, 4	• Millorar els resultats trobats fins ara.	• No s'ha dedicat temps a fer els mètodes de ensambles. Pel

	○ Utilitzar per cada classificador el mètode Boosting ○ Utilitzar per cada classificador el mètode Bagging. • Seguir treballant en l'informe de progrés 1.	que he vist, no aporten gaire millora a aquest tipus de prediccions. • S'ha aprofitat la setmana per treballar en l'informe de progrés 1 i deixar-lo al dia. • S'ha automatitzat el procés de guardar resultats, mostrar gràfics i similar. • Es treballa una mica en Hyper Paramter Tuning (previst per la primera setmana de novembre)
Novembre, 1	• Enllestir informe progrés 1 • Ratificar planificació (si escau) • Proposar canvis (si escau) • Millorar els resultats obtinguts fins al moment utilitzant els mètodes de Hyper Parameter Tuning. ○ Aplicar Grid Search ○ Aplicar Random Search ○ Aplicar Bayesian Optimization	• S'ha acabat de realitzar l'informe de progrés 1. • S'ha millorat la feina feta a HyperParamter Tuning. Ara, a part de tenir els resultats, es fan plots i es mostren els millors paràmetres i tal • S'ha dedicat bastant temps a refer alguns gràfics en el format correcte per tal que sigui més fàcil de llegir.

Iteració 3

L'objectiu d'aquesta iteració és utilitzar els mètodes més recents basats en Deep Learning i obtenir resultats de les prediccions de depressió a les xarxes socials. Per una banda es començarà utilitzant Recurrent Neural Networks, explorant els models que tenen més èxit i com es comporten amb el sentiment analysis.

Per altra banda també s'utilitzarà el mètode més recent i que ara mateix està sent el que té més èxit, les anomenades xarxes Transformer. Es posaran a prova els models que hi estan basats i es veurà quin resultat obtenen, comparant-los tant amb les RNN com amb els mètodes tradicionals.

Novembre, 2	• Segona sessió de seguiment • Instal·lar llibreries necessàries per aquesta iteració. • Aplicar la xarxa neuronal LSTM • Seguir treballant en l'informe de progrés 2	
Novembre, 3	• Aplicar la xarxa neuronal GRU • Seguir treballant en l'informe de progrés 2	
Novembre, 4	• Aplicar la xarxa Transformers GPT	

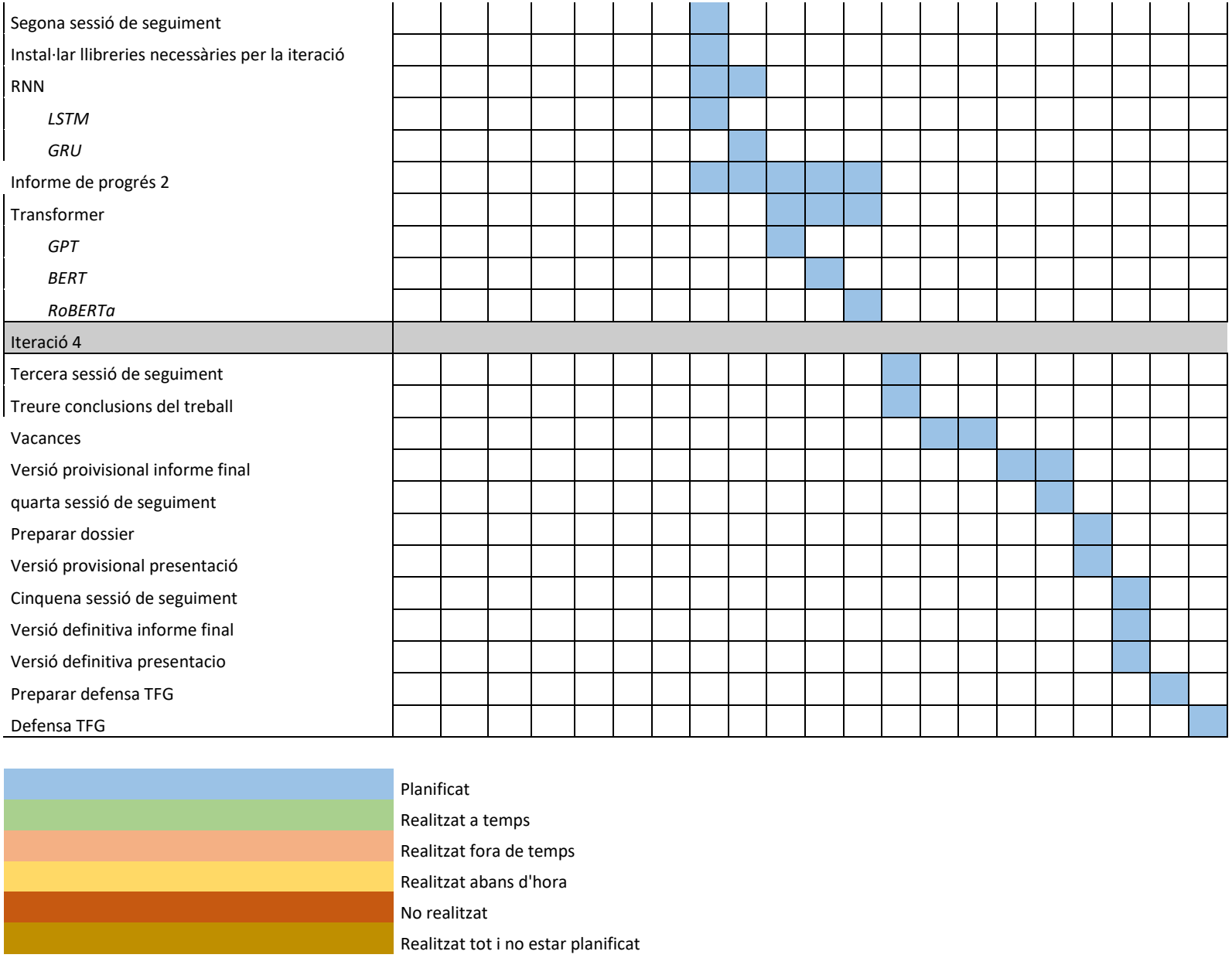
	• Seguir treballant en l'informe de progrés 2	
Desembre, 1	• Aplicar la xarxa Transformers BERT • Seguir treballant en l'informe de progrés 2	
Desembre, 2	• Aplicar la xarxa Transformesrs Ro-BERTa • Finalitzar l'informe de progrés 2	

Iteració 4

L'objectiu d'aquesta iteració ja no és seguir desenvolupant, sinó treure conclusions del que s'ha fet fins al moment i preparar tot el material per la presentació i l'entrega de l'informe final. També caldrà deixar ordenat el dossier del treball.

Desembre, 3	• Tercera sessió de seguiment • Treure conclusions del treball	
Desembre, 4	• Vacances	
Gener, 1	• Vacances	
Gener, 2	• Fer versió provisional de l'informe final	
Gener, 3	• Quarta sessió de seguiment • Fer versió provisional de l'informe final	
Gener, 4	• Preparar dossier • preparar presentació provisional	
Febrer, 1	• Cinquena sessió de seguiment • Corregir versions provisionals i deixar-les com a finals	
Febrer, 2	• Preparar defensa TFG	
Febrer, 3	• Defensa TFG	

Setmanes	Setembre, 3	Setempre, 4	Octubre, 1	Octubre, 2	Octubre, 3	Octubre, 4	Novembre, 1	Novembre, 2	Novembre, 3	Novembre, 4	Desembre, 1	Desembre, 2	Desembre, 3	Desembre, 4	Gener, 1	Gener, 2	Gener, 3	Gener, 4	Febre, 1	Febrer, 2	Febrer, 3
Iteració 1																					
Reunió inicial																					
Informe inicial																					
Instal·lar LaTex																					
Iteració 2																					
Primera sessió de seguiment																					
Buscar informació i estat de l´art																					
Definir objectius																					
Detallar planificació i tasques																					
Preparar entorn de treball																					
Instal·lar Python i IDE																					
Instal·lar llibreries necessàries per la iteració																					
Exploratory Data Analysis																					
Classificadors Tradicionals																					
Classificador: Naïve Bayes																					
Classificador: Decision Tree																					
Classificador: Random Forest																					
Classificador: Support Vector Machine																					
Classificador: K Nearest Neighbour																					
Treballar en l'informe de progrés 1																					
Millorar resultats																					
Boosting																					
Bagging																					
Retocar feina anterior no realitzada correctament																					
Eliminar Datasets no útils. Buscar-ne de nous																					
Hyper Parameter Tuning																					
Optune framework																					
ratificar planificació (si escau)																					
Proposar canvis (si escau)																					
Iteració 3																					


Segona sessió de seguiment
Instal·lar llibreries necessàries per la iteració
RNN
LSTM
GRU
Informe de progrés 2
Transformer
GPT
BERT
RoBERTa
Iteració 4
Tercera sessió de seguiment
Treure conclusions del treball
Vacances
Versió proivisional informe final
quarta sessió de seguiment
Preparar dossier
Versió provisional presentació
Cinquena sessió de seguiment
Versió definitiva informe final
Versió definitiva presentacio
Preparar defensa TFG
Defensa TFG

Planificat
Realitzat a temps
Realitzat fora de temps
Realitzat abans d'hora
No realitzat
Realitzat tot i no estar planificat

BIBLIOGRAFIA

[1] Yasar, K. (2022, 12 abril). *social networking*. WhatIs.com. Recuperado 2 de octubre de 2022, de https://www.techtarget.com/whatis/definition/social-networking

[2] boyd, D. M. & Ellison, N. B. (2007, octubre). Social Network Sites: Definition, History, and Scholarship. *Journal of Computer-Mediated Communication*, *13*(1), 210-230. https://doi.org/10.1111/j.1083-6101.2007.00393.x

[3] Chan, T. K. H., Cheung, C. M. K., & Lee, Z. W. Y. (2020, December 5). *Cyberbullying on social networking sites: A Literature Review and future research directions*. Information & Management. Retrieved October 2, 2022, from https://www.sciencedirect.com/science/article/pii/S0378720620303499

[4] Rosenquist, J. N. (2010, 16 marzo). *Social network determinants of depression*. Nature. Recuperado 2 de octubre de 2022, de https://www.nature.com/articles/mp201013

[5] Dey, L. (2016, 31 octubre). *Sentiment Analysis of Review Datasets Using Naive Bayes and K-NN Classifier*. arXiv.org. Recuperado 5 de octubre de 2022, de https://arxiv.org/abs/1610.09982

[6] *Sentiment analysis on Twitter data using KNN and SVM - semantic scholar*. (n.d.). Retrieved October 5, 2022, from https://pdfs.semanticscholar.org/05a8/78000170abcd0c6f8208080470858422e17c.pdf

[7] *Sentiment analysis on Twitter data using KNN and SVM - semantic scholar*. (n.d.). Retrieved October 5, 2022, from https://pdfs.semanticscholar.org/05a8/78000170abcd0c6f8208080470858422e17c.pdf

[8] https://www.kaggle.com/code/ardawrld/twitter-sentiment-analysis-about-the-depression/data

[9] https://www.kaggle.com/code/mpwolke/depression-sentiment-analysis-classifiers/data

[10] https://www.kaggle.com/datasets/infamouscoder/depression-reddit-cleaned

[11] https://www.kaggle.com/datasets/gargmanas/sentimental-analysis-for-tweets

[12] Claesen, M., & De Moor, B. (2015, April 6). *Hyperparameter search in machine learning*. arXiv.org. Retrieved October 5, 2022, from https://arxiv.org/abs/1502.02127

[13] Bernardo, M., Alberto, L., & Simone, M. (2020, May 18). *Comparing machine learning and deep learning approaches on NLP tasks for the Italian language*. IRIS. Retrieved October 6, 2022, from https://cris.fbk.eu/handle/11582/322156

[14] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017, December 6). *Attention is all you need*. arXiv.org. Retrieved October 6, 2022, from https://arxiv.org/abs/1706.03762

[15] Kokab, S. T., Asghar, S., & Naz, S. (2022, April 10). *Transformer-based deep learning models for the sentiment analysis of social media data*. Array. Retrieved October 6, 2022, from https://www.sciencedirect.com/science/article/pii/S2590005622000224

[16] *Overview of the Transformer-based models for NLP tasks*. IEEE Xplore. (n.d.). Retrieved October 6, 2022, from https://ieeexplore.ieee.org/abstract/document/9222960/

[17] *LSTM and GRU neural network performance comparison study: Taking Yelp Review dataset as an example*. IEEE Xplore. (n.d.). Retrieved October 6, 2022, from https://ieeexplore.ieee.org/document/9221727

[18] Pimpalkar, A. P. (2020). *Influence of Pre-processing Strategies on the Performance of ML Classifiers Exploiting TF-IDF and BOW Features*. Retrieved from https://revistas.usal.es/index.php/2255-2863/article/download/ADCAIJ2020924968/24569/

[19] Networks, T. A. P., Akiba, T., Networks, P., Networks, (2019, July 1). *Optuna: A Next-generation Hyperparameter Optimization Framework*. ACM Conferences. Retrieved November 1, 2022, from https://dl.acm.org/doi/abs/10.1145/3292500.3330701