UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona

FIB

EXPLORACIÓ DE ZERO-SHOT I FEW-SHOT LEARNING PER A EXTRACCIÓ D'INFORMACIÓ DE TEXTOS MÈDICS

XENIA CALISALVO VECIANA

Director/a: LLUIS PADRO CIRERA (Departament de Ciències de la Computació)

Titulació: Grau en Enginyeria Informàtica (Computació)

Memòria del treball de fi de grau

Facultat d'Informàtica de Barcelona (FIB)

Universitat Politècnica de Catalunya (UPC) - BarcelonaTech

23/01/2024

Abstract

The main objective of this project is to use Large Language Models (LLM) to carry out the extraction of information from medical texts using learning techniques such as zero-shot and few-shot learning.

The information from these medical texts will come from the DDI corpus dataset, composed of the Drug Bank and MEDLINE sets, which consists of a semantically annotated corpus of documents describing interactions between drugs. From these data, we will extract information to identify the different pharmacological substances present in these texts using smaller-sized language models such as GPT2 and GPTNeo. With these models, an exploration of the different complexities available will also be conducted to study the computational resources required for predictions and the time needed for these tasks. To use these techniques and models, it is necessary to have GPUs and CPUs with a certain power and their specifications should remain consistent across different experiment reproductions to maintain reproducibility. Therefore, we will rely on the Google Colab platform, which provides a simple and intuitive environment for conducting our research.

Resum

L'objectiu principal d'aquest projecte és fer ús dels models massius de llenguatge (*Large Language Model* - LLM) per dur a terme l'extracció d'informació de textos mèdics fent ús de tècniques d'aprenentatge com *zero-shot i few-shot learning*.

La informació d'aquests textos mèdics provindrà del conjunt de dades *DDI corpus*, compost pels conjunts *Drug Bank* i *MEDLINE*, que consisteix en un corpus semànticament anotat de documents que descriuen interaccions entre medicaments. A partir d'aquestes dades extraurem la informació per trobar les diferent substàncies farmacològiques que es troben en aquests textos utilitzant models massius de llenguatge de mida reduïda, com poden ser GPT2 i GPTNeo. Amb aquest models es farà també una exploració de les diferents complexitats disponibles per tal de realitzar un estudi dels recursos computacionals necessaris per a les prediccions i del temps necessari per aquestes. Per a fer ús de les tècniques i models és necessari l'ús de GPUs i CPUs amb certa potencia i que a la vegada les seves especificacions no variïn entre diferents reproduccions dels experiments per tal de mantenir la reproductibilitat d'aquests. Per això, ens recolzarem de la plataforma Google Colab que ens proporciona un entorn senzill i intuïtiu per dur a terme la nostre investigació.

Resumen

El objetivo principal de este proyecto es utilizar los modelos masivos de lenguaje (*Large Language Model* - LLM) para llevar a cabo la extracción de información de textos médicos mediante el uso de técnicas de aprendizaje como el *zero-shot* y el *few-shot learning*.

La información de estos textos médicos provendrá del conjunto de datos *DDI corpus*, compuesto por los conjuntos *Drug Bank* y *MEDLINE*, que consiste en un corpus semánticamente anotado de documentos que describen interacciones entre medicamentos. A partir de estos datos, extraeremos la información para encontrar las diferentes sustancias farmacológicas presentes en estos textos utilizando modelos masivos de lenguaje de tamaño reducido, como GPT2 y GPTNeo. Con estos modelos, también se realizará una exploración de las diferentes complejidades disponibles para llevar a cabo un estudio de los recursos computacionales necesarios para las predicciones y del tiempo requerido para estas. Para utilizar estas técnicas y modelos, es necesario el uso de GPUs y CPUs con cierta potencia y que, a su vez, sus especificaciones no varíen entre diferentes reproducciones de los experimentos para mantener la reproducibilidad de estos. Por ello, nos apoyaremos en la plataforma Google Colab, que nos proporciona un entorno sencillo e intuitivo para llevar a cabo nuestra investigación.

Índex

1 Introducció i contextualització

L'objectiu d'aquest document és definir d'una forma clara i precisa tots els factors relacionats amb la gestió del Treball de Fi de Grau (o altrament TFG) dins la modalitat A. Concretament, farem referència a el context i la motivació del problema a resoldre, el temps i els costs associats a resoldre'l.

Primerament, començarem introduint-nos amb el context en el qual esta ubicat el nostre treball i parlarem sobre els conceptes claus que son necessaris per entendre aquest projecte.

1.1 Context

El treball que tenim entre mans es centra en la utilització dels models massius de llenguatge o *Large Language Model* - LLM. Aquests son algoritmes d'aprenentatge profund que poden realitzar una gran varietat de tasques relacionades amb el llenguatge natural processat (NLP), concretament, el seu objectiu principal és el de comprensió i manipulació de textos.

Gràcies als avenços i la divulgació de la intel·ligència artificial els últims anys i amb l'aparició de chatbots, models de llenguatge fets a partir de IA que son capaços de generar text a partir d'anterior conversacions com el conegut ChatGPT, ens trobem que aquest tipus d'algoritmes cada dia son mes presents en el nostre dia a dia. El que es creu que és la clau d'aquest èxit és el nivell d'interacció dinàmica i el fet de que cada cop estan disponibles més bases de dades públiques fent que aquest tipus d'IAs sigui capaç d'adaptar-se més fàcilment a tot tipus de situacions específiques i utilitzades en tot tipus de sectors. Volem remarcar que la funcionalitat dels LLM van més enllà que un simple xat de pregunta resposta, quan aconseguim entrenar aquests amb les dades necessàries i correctes la interacció amb aquest tipus de models d'IA s'impulsa molt més, aconseguint la informació d'experts que moltes empreses o/i organitzacions necessiten, d'una forma molt més òptima.

Aquest és un treball d'investigació compres en el marc de la Facultat d'Informàtica de Barcelona - FIB, concretament de l'especialitat de Computació i centrat en l'anàlisi de dades i aprenentatges automàtic, on el que volem és aconseguir extreure informació de textos de caràcter medic. L'objectiu principal és el d'aconseguir classificar medicaments tant grupalment com concretament i diferenciar-los de les drogues no aptes per al consum humà. Per això, utilitzarem diferents models d'aprenentatge automàtic, els quals han estat introduïts en assignatures com Aprenentatge automàtic o Mineria de dades, per analitzar quins d'ells obtenen millors resultats.

1.2 Conceptes

A continuació introduirem alguns conceptes claus que ens serà útil tenir en compte d'ara en endavant per tal d'entendre millor el problema i les solucions proposades.

1.2.1 Intel·ligència artificial

La intel·ligència artificial es pot definir com un conjunt d'algoritmes que a mesura de que recopilen informació son capaços de solucionar millor el problema a tractar, intentant recrear d'aquesta manera el que faria la intel·ligència humana. És a dir, automatitza l'aprenentatge i el descobriment repetitiu a traves d'informació en forma de dades.

Hi ha moltes formes d'intel·ligència artificial diferents que van des de la intel·ligència artificial clàssica, la qual consisteix en l'anàlisi formal i estadístic del comportament humà davant de diferents problemes, fins a diferents formes d'aprenentatge automàtic com per exemple les complexes xarxes neuronals artificials les quals estan inspirades en el funcionament del cervell humà imitant amb capes de nodes connectades entre elles el que simularia el funcionament de les neurones.

Actualment la majoria d'exemples que tenim d'aquest camp, des de ordinadors capaços de jugar per si mateixos als escacs a cotxes de conducció autònoma, consisteixen bàsicament en models d'aprenentatge profund o/i al processament de llenguatge natural.

1.2.2 Aprenentatge automàtic

L'aprenentatge automàtic, o *machine learning* en anglès, consisteix en el desenvolupament de teories, tècniques i algorismes que permetin explorar mètodes automàtics per inferir models a partir de dades. Per tal d'arribar-hi s'utilitzen diferents disciplines com la informàtica, l'estadística multi-variant i l'optimització matemàtica d'entre altres.

Per tal d'arribar a la solució d'aquests problemes utilitzem un conjunt d'entrenament, el qual son les dades que li proporcionem al nostre model per tal de que aprengui a partir d'informació, característiques i etiquetes de les classes, i un conjunt de test que ens permet validar la informació que el nostre model ha apres comprovant quin percentatge d'encert i error aconsegueix.

En aquest treball ens centrarem en aquest tipus d'intel·ligència artificial, concretament en els dos tipus que explicarem a continuació *Zero-shot learning* i *Few-shot learning*.

1.2.3 *Zero-shot learning*

Zero-shot learning és un model d'aprenentatge automàtic, més concretament d'aprenentatge profund (*deep learning*) que consisteix en predir classes sense haver vist aquestes classes en el conjunt d'entrenament [1].

Una bona forma d'imaginar el comportament d'aquest es partir de la base que tenim l'objectiu de classificar tots i cada un dels tipus d'animals que existeixen. Segons el que tenim entès de com funciona l'aprenentatge automàtic fins ara, el primer que se'ns vindria al cap es pensar que necessitem un conjunt de dades que com a mínim contingues un exemple etiquetat de cada tipus d'animal. Tinguen en compte que al mon existeix gairebé 2 milions d'animals diferents ens adonem de que aquesta idea tampoc és gaire factible o com a mínim molt ineficient.
Per aquest tipus de casos és quan se'ns ve a la ment la idea d'intentar disminuir la dependència entre classes etiquetades. D'aquesta forma és com apareix la idea del *Zero-shot learning*.

Com en els models tradicionals supervisats, segueixen existint dos fases; la d'entrenament i la de test. Però ha diferencia d'aquestes, en la fase d'entrenament el model aprèn sobre les classes a partir de llegir textos que corresponen a aquestes. Després d'un entrenament amb molts més textos el model hauria de ser capaç de trobar patrons en els textos que es relacionen amb les classes i extrapolar-ho a la classificació de noves tasques.

1.2.4 *Few-shot learning*

De la mateixa manera que el *Zero-shot learning*, el *Few-shot learning* és un model d'aprenentatge profund on el conjunt d'entrenament té una informació limitada. Aquest mètode és útil sobretot quan els exemples d'entrenament son difícils de trobar o quan el cost d'aquesta informació és molt alt [2].

Per tal de dur a terme aquest model s'utilitza *N-way-K-shot classification*, on cada tasca inclou N classes amb K exemples. A més, cap de les tasques pot tenir classes d'altres tasques. En el paradigma clàssic, quan tenim una tasca, el algoritme està aprenent si aquesta tasca millora amb la experiència. En canvi, aquí comptem amb un conjunt de tasques i l'algoritme esta aprenent a millor amb l'experiència i el nombre de tasques. Aquest tipus d'aprenentatge s'anomena meta *learning*.

1.2.5 *Natural language processing*

Natural language processing o llenguatge natural processat és el que utilitza l'aprenentatge automàtic per tal de trobar la estructura i el significat dels texts. Aquest, com veiem a la Figura 1, esta constituït en part per disciplines com les ciències de la computació, el llenguatge humà i l'intel·ligència artificial.

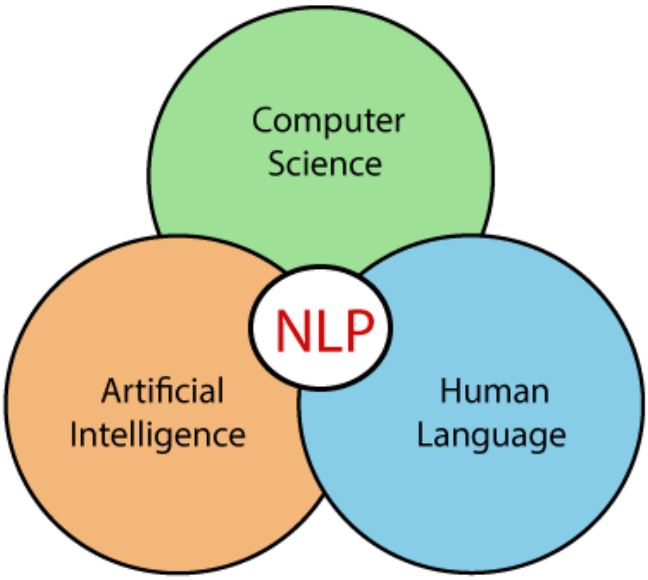

Figura 1: Diagrama que mostra la relació entre les diferents disciplines per constituir els NLP

Gràcies a això podem extreure tot tipus d'informació traduir, classificar i fins i tot corregir faltes d'ortografia dels texts.

1.2.6 *Large Language Model*

Un *Large Language Model* o LLM és un algoritme d'aprenentatge profund que pot dur a terme una gran varietat de tasques de llenguatge natural [3]. Aquest tipus de model utilitza conjunts de dades enormes i s'utilitzen per tal de reconèixer, traduir, predir o generar text o altre contingut. igual que els models anteriors, aquests han de ser pre-entrenats i testejats per tal de funcionar correctament.
Les aplicacions d'aquest models son gairebé infinites, des de ciències de la salut, finances, programació, assistents AI, els actuals chatbots, i això és només una petita llista.

Els LLM estan composts per múltiples capes de xarxes neuronals:

- ***The embedding layer:*** crea *embeddings* des del text d'entrada. Per fer-ho captura el significat semàntic i sintàctic de la entrada per tal de que el model pugui entendre el context.
- ***The feedforward layer (FFN):*** esta feta a partir de moltes capes connectades entre elles que transformen les entrades. Aquesta capa té permès fer abstraccions d'alt nivell.

- ***The recurrent layer:*** interpreta les seqüencies de paraules en el text d'entrada i captura les relacions entres les paraules d'una frase.
- ***The attention mechanism:*** permet al model de llenguatge centrar-se en parts individuals del text d'entrada que son rellevant per la tasca que estem intentant resoldre. Aquesta capa és la que permet al model de generar les sortides més precises.

1.3 Identificació del problema

Els models massius de llenguatge son un tipus de models molt útils avui en dia i que ens permeten realitzar una gran quantitat de tasques diferents relacionades amb el mon de la intel·ligència artificial. Com ja hem comentat anteriorment, la resolució de tasques que poden realitzar és multi disciplinar i compren des del àmbit de la programació, a productes de finances fins a l'ajuda en àmbits de la salut. En el nostre cas ens volem centrar en aquesta última temàtica, concretament amb la extracció d'informació dels textos mèdics.

Podem trobar-nos amb un munt d'exemples de textos mèdics diferents, des de receptes de medicaments, exàmens de la facultat de medicina, simptomatologia d'una malaltia o la simple descripció de com es sent un pacient que ha anat al centre d'atenció primaria per a una revisió. De tots aquests textos es pot extreure molt informació útil que al relacionar-se entre si ens pot ajudar, a per exemple, descobrir relacions entre medicaments i efectes secundaris, processar massivament articles científics i detectar descobriments interessants que algú ha escrit, realitzar estadístiques d'informes hospitalaris o d'altres.

Amb això ens trobem principalment davant dels següents problemes:

- **Eficiència:** els LLM solen ser models amb una quantitat d'informació gegantina, el que provoca que sigui bastant ineficient el fet d'entrenar-los i que requereixi bastant de temps.
- **Falta de classes per a l'entrenament:** un altre problema amb el que ens trobem és que d'alguns medicaments es pot trobar molt poques dades fins al punt que el nombre de mostres que existeixi per al conjunt d'entrenament sigui molt petit o inexistent.
- **La naturalesa crítica de la solució:** al tractar-se de temes de caràcter mèdic la solució d'aquests pot ser molt crítica necessiten uns nivells d'encert molt elevats per tal de poder ser utilitzats.

En conjunt veiem que la tasca necessita d'unes solucions d'alta qualitat les quals parteixen d'una prèvia comprensió medica, que ens vindrà donada per el processament massiu d'articles mèdics i científics.

1.4 Agents implicats

Al projecte trobem diversos agents implicats:

- **El personal del projecte:** Som els que desenvoluparem tot el projecte en el temps preestablert per això i complint les expectatives d'aquest.
- **Els malalts:** Les persones les quals pateixen malalties de les que posteriorment s'escriuen informes mèdics que utilitzarem per descobrir relacions entre els medicaments i els seus efectes secundaris.
- **Hospitals:** Contenen informes hospitalaris dels quals podrem fer control de costos i de prescripcions per controlar la quantitat subministrada i els tipus d'aquests. En aquest apartat podem relacionar tant directors dels hospitals, polítics com el personal sanitari que hi treballi ja que seran els principals beneficiaris d'aquest tipus d'informació.
- **Investigadors:** Investigadors que prèviament ja han realitzat estudis sobre aquest tema i dels quals farem ús de les seves investigacions per tal de trobar descobriments a tenir en compte.

2 Justificació

2.1 Estudi de solucions existents

Com el mon de la intel·ligència artificial i el aprenentatge automàtic esta cada cop més a l'alça podem trobar una gran varietat de solucions i recursos enfocats en la optimització i possible aplicació del LLMs. En el nostre cas ens volem fixar concretament en aquells LLMs amb la capacitat per a interpretar i estudiar textos mèdics, dels quals tot i que la llista és més reduïda, també existeixen.

Pathways Language Model, o PaLM, és un LLM de 540 bilions de paràmetres que utilitza un conjunt d'estratègies amb demandes que esta entrenat per tal de contestar a preguntes relaciones amb el context mèdic [4]. Aquest LLM aconsegueix un 67'6% de precisió en MedQA, l'examen per a la llicencia mèdica d'Estats Units, un resultat que supera l'aprovat però que no és del tot encoratjador tinguen en compte que a més son respostes relativament senzilles comparades amb casos pràctics de veritat.

SoTA és un altre LLM que s'ha utilitzat en la resolució de preguntes de caràcter mèdic. Aquest consisteix en convertir preguntes de llenguatge natural en consultes d'SQL. Aquest model utilitza 15B paràmetres per a la generació de les tasques aconseguint molt bons resultats en la majoria de models. Malgrat això, a l'hora d'utilitzar-se en l'enfocament mèdic no aconsegueix millorar els resultats de l'anterior *Pathways Language Model*, i es queda amb un 50'3% d'encert en el MedQA, veieu Figura 2.

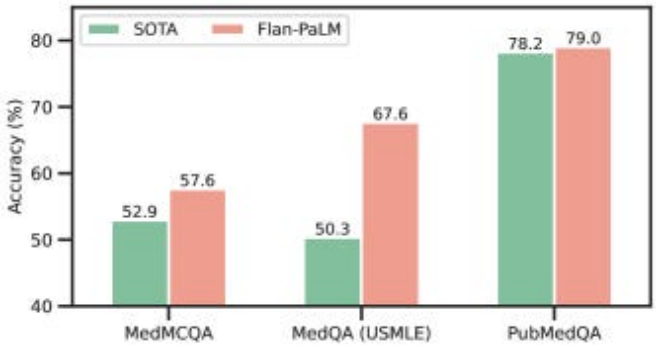

Figura 2: Gràfic comparatiu entre els models SOTA i PaLM

Com podem veure els resultats d'aquests models no son suficientment òptims ni tenen una precisió suficient com pel que es demana en aquest tipus de problemes cosa que ens fa creure que si és interessant el investigar una nova solució o el que creiem que és encara millor, adaptar una solució ja existent per aconseguir uns millors resultats.

2.2 Eines

Per al desenvolupament de la nostre aplicació necessitarem un conjunt d'eines *software*. En aquest moment hi ha infinitat de diferents recursos disponibles a internet dels quals podem disposar. La primera limitació és que molts d'aquests recursos son de subscripció o pagament i per tant haurem de centrar-nos només en les eines de codi obert o/i de les quals contem amb la llicencia.

2.2.1 GPT-J, 6B

GPT-J de 6B és el model principal d'LLM que utilitzarem al projecte, on 6B fa referència el nombre de paràmetres entrenables. La diferència principal entre aquest model i el conegut GPT-3 és la diferència de paràmetres, que mentre aquest primer conta amb 6 bilions, el segon esta sobre els 175 bilions [5]. El principal avantatge d'aquest davant de GPT-3 és que el codi d'aquest esta disponible per tothom per tal de modificar-lo i per tant per fer-lo més precís per a les tasques. S'ha comparat aquest llenguatge amb altres també de codi obert com Claude 2 o Truca 2 però ens hem decantat per aquest ja que és molt més complert i esta molt més documentat, fet que ens facilitara fer proves amb ell. Aquest és un llenguatge casual el qual depèn completament de les prèvies respostes i el context donat amb anterioritat ja que a cada pas utilitza la predicció anterior per tal de crear la nova predicció.

El factor crític de com de precisos son els LLM és el previ entrenament d'aquests. Un exemple clar és com de bé funciona ChatGPT ja que ha estat entrenat usant GPT-3 però fent-ho específic per a les preguntes i respostes que fem els humans. D'aquesta manera, com ha estat entrenat específicament, funciona de molt millor manera que no altres models molt més grans però que no han estat entrenats per un tema en concret.

Per aquest tema GPT-J ha estat entrenat amb 825 gigabytes que pertanyen a 22 conjunt de dades de qualitat de diferents fonts els quals li permeten tenir bon resultats en una quantitat considerable de dominis. Concretament el model consisteix en 28 capes amb un a dimensió del model de 4096 i un avens de dimensions de 16384. Amb la estructura i les estadístiques de text d'aquests conjunt de dades els paràmetres aprenen a mimetitzar-se amb la semàntica del text. A conseqüència directe d'això aconseguim discernint patrons abstractes del text de la mateixa manera que les persones ho fan, donant-li capacitats per moltes tasques de llenguatge natural sense tenir gaire informació.

Com hem pogut anar veient GPT-J és una bona opció per tal de modificar un LLM en comptes de tractar de crear una nova solució des de zero i a més ens permet ajustar-lo en un context en específic.

2.2.2 Google Colab

Servei gratuït de Google que s'utilitza per tal d'executar codi *Python* al nuvol. Utilitza l'entorn de *Jupyter Notebook* i no és necessari d'instal·lar cap *software*, cosa que el fa bastant atractiu per comoditat. Un altre factor decisiu per triar-lo és la capacitat d'utilitzar el seu accelerador per GPU. Tot i que no és el més potent, ja que és compartit, és bastant útil i dona un millor servei que una CPU convencional per a tasques d'aprenentatge automàtic.

2.2.3 Clúster del laboratori de càlcul de la UGDSI

Se'ns ha proporcionat accés al clúster del laboratori de càlcul de la UGDSI per tal de poder utilitzar bases de dades més gran per entrenar el nostre model que les que podríem utilitzar en el nostre propi ordinador. Si que és cert que en certa manera aquestes bases de dades també podrien ser carregades des d'un ordinador de sobre taula o portàtil amb unes especificacions més pobres però la quantitat tant gran de dades que suposaria això ens faria perdre una quantitat important del temps de treball i no seria òptim a l'hora d'experimentar amb les diferents mides d'informació que volem provar en el projecte.

3 Definició de l'abast

Una de les parts més importants d'un projecte és la de definir concretament a on volem arribar i perquè hi volem arribar. De la mateixa manera, també és important parlar dels objectius i requeriments que es necessitaran per arribar-hi. En aquest apartat tractarem tot això e intentarem veure quines parts del projecte podrien perillar i com solucionar aquests possibles riscs.

3.1 Objectius

L'objectiu principal d'aquest projecte és el de classificar i reconèixer drogues i medicaments a partir de la informació de textos mèdics fent ús de models de llenguatge de mida reduïda, com GPT-J (6B), en tasques d'extracció d'informació, comparant-ne el rendiment i el cost d'entrenament amb diferents mides d'exemples d'entrenament. Concretament podem parlar dels següents sub-objectius per tal de que el treball sigui un èxit:

1. **Selecció de dades i preparació.** Recopilarem conjunts de dades rellevants per a tasques d'extracció d'informació que es puguin utilitzar en l'avaluació. També pre-processarem les dades, incloent-hi neteja, segmentació i etiquetatge, per preparar-les per a l'entrenament i l'avaluació.

2. **El reconeixement i la classificació de les drogues i medicaments.** Utilitzarem els algoritmes de *Zero-shot learning* i *Few-shot learning* juntament amb LLM, explorant els diferents hiperparàmetres i configuracions, per tal de desenvolupar un model capaç de classificar tant en grups com individualment els diferents tipus de drogues. Haurem d'aconseguir que els resultats siguin eficients i que la precisió d'aquests arribi a uns valors òptims.

3.2 Requeriments

Els requeriments que haurem de complir en tot moment de la experimentació de manera que els resultats siguin útils son els següents:

1. La **informació** que utilitzarem per tal d'entrenar els models haurà de ser **correcte, fiable i contrastada**. És molt important ja que com els models aprenen de les iteracions anteriors un error en el inici seria un error fatal per al projecte.

2. No sobrepassar certs límits en quant a la **mida dels LLM** a treballar. Ja que des d'un principi hem decidit que els LLM amb els quals treballaríem consistirien en llenguatges de mida reduïda haurem de vigilar que això continuí sent així durant la experimentació. D'un altre manera la comparativa entre diferents llenguatges no seria justa ja que la mida d'aquests llenguatges afecta sobre les seves capacitats de precisió al completar les tasques i estaríem donant-li més pes a la mida que no a les opcions que

configuréssim o a tota la feina darrera de fer-los més específics per al nostre tipus de problema.

3. La **eficiència**. La execució dels models ha de ser eficient per poder obtenir resultats en un temps no molt elevat. El motiu principal de la classificació dels medicaments és aconseguir millorar la eficiència respecte al que tardarien persones físiques així que si el temps no és optim perdem la motivació per a la classificació automàtica.

4. La **escalabilitat**. És important no sobre ajustar-nos als conjunts d'entrenament per tal de poder utilitzar diferents conjunts de dades i seguir obtenint bons resultats.

3.3 Obstacles i riscs

Els possibles obstacles i riscs que podríem plantejar serien infinits però ja que hi ha diferents nivells de probabilitats de que succeeixin entre ells ens centrarem amb els més comuns i que sabem com podem evitar o com redreçar el projecte:

1. **Inexperiència.** Com a tot projecte de final de grau ens trobem davant d'un alumne amb inexperiència davant de la tasca que se li proposa i que haurà d'afegir un esforç extra per tal de suplir aquesta manca de coneixements amb la que es trobara.

2. **Rendiment.** Els LLM poden arribar a ser d'una mida descomunal i entrenar-ne un suposaria una quantitat de temps de la qual no disposem. Justament per aquesta raó és per la qual ens centrarem en LLM de mida reduïda. De la mateixa manera s'haurà de vigilar amb els conjunt d'entrenament a utilitzar.

3. **Gestió del temps.** En el quadrimestre que l'autor del treball esta realitzant el TFG esta cursant també un altre assignatura amb una gran carrega de treball i buscant ofertes laborals. Per tot això pot ser que hi hagi moments en els quals els pics de feina siguin més elevats que en altres i suposin un risc per al projecte.

4. **Localització actual.** Actualment l'autor del treball viu a més de 2 hores de camí de la facultat ja que no està situat a Barcelona i els desplaçament fins aquesta poden suposar un risc tant en el temps que implicarien com per la possible impossibilitat de realitzar-se així que majoritàriament la comunicació amb el personal del centre relacionada amb el projecte es farà de manera telemàtica.

5. **Desconeixement del funcionament del clúster.** L'autor del treball no ha treballat mai amb el clúster del qual se li ha donat accés i per tant haurà d'experimentar el funcionament i les capacitats d'aquest abans de començar a utilitzar-lo en el projecte suposant un augment d'hores indefinit en quant a feina.

6. **Accidents i/o malalties.** Tot i que no s'espera cap tipus d'accident mai es pot estar del tot segur així que no esta de més tenir-lo contemplat, de la mateixa manera que agafar alguna malaltia que entorpeixi el correcte funcionament del projecte.

7. **No correspondència dels resultats.** Durant tota la experimentació pot ser que alguns del resultats no quadrin amb la teoria i per tant s'haurà d'investigar el motiu i solucionar-lo afegint més temps del previst amb aquest tema.

8. **Problemes amb utilització de dispositius o internet per al treball.** Degut a que l'autor del treball en aquests moments viu en una casa on no es disposa de wifi, hi ha una impossibilitat d'instal·lar-ho, ha hagut de moure el seu despatx a una casa aliena on poden haver-hi problemes externs que ell no controli i provoquin un augment de les hores o una impossibilitat de treball en algun moment concret.

4 Metodologia

Totes les metodologies tenen en comú que son conjunts de tècniques i eines que s'utilitzen per planificar, dissenyar, desenvolupar, implementar, i mantenir un software. En un projecte és molt important escollir un tipus de metodologia dintre de totes les possibles a dur a terme per tal de que el desenvolupament de software segueixi una estructura i coherència a la vegada que es pugui testejar per tal de trobar errors i corregir-los.

4.1 Metodologia triada

En el nostre projecte utilitzarem una metodologia del tipus àgil. Aquestes s'enfoquen en la iteració i en entregar codi en terminis més curts ja que així estem més preparats per si els requisits del treball canvien durant el procés, és a dir, és més flexible [6]. Els cicles de desenvolupament que seguirem seran d'aproximadament una setmana, suficientment curt com per formar part d'aquesta metodologia, on es dissenyaran, implementaran i provaran algunes funcionalitats. En aquests cicles, veieu Figura 3, dedicarem cada setmana a realitzar les diferents versions dels models, documentat cada pas d'aquests i un cop finalitzats també documentarem els resultats obtinguts.

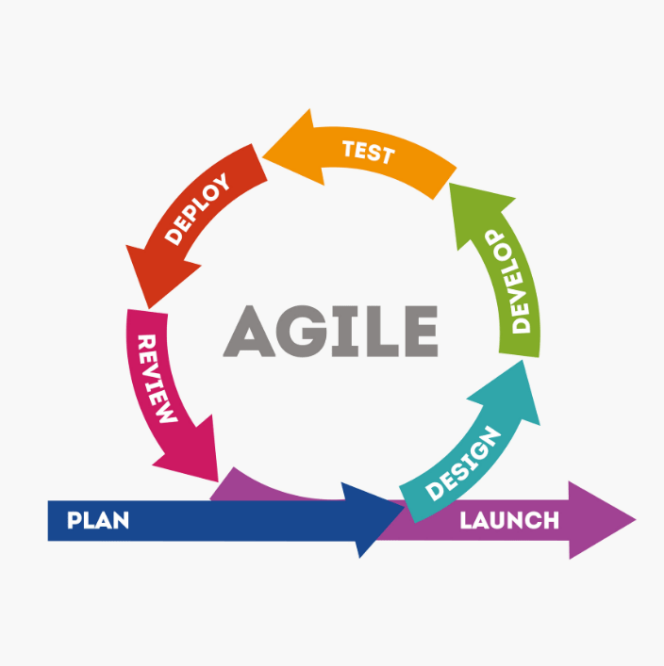

Figura 3: Principis de la metodologia àgil

Les reunions amb el director el projecte es faran o setmanalment, coincidint

amb els cicles de treball o cada cop que es compleixi un dels objectius del treball per tal de poder validar-los i continuar cap endavant. D'aquesta manera aconseguim tractar cada petit conflicte a temps i resoldre'l ràpidament abans de que alenteixi el treball.

El desenvolupament del treball es farà principalment des del despatx personal de l'autor del projecte, havent-hi possibles excepcions que l'obliguin a reunir-se presencialment a la facultat.

4.2 Mètode de validació

El que ens funcionara com a mètode de validació principal serà bàsicament els resultats de la experimentació. Primerament, com es tracten de models d'aprenentatge automàtic sempre contarem amb els conjunt de test per tal de validar els resultats i veure la precisió que estem obtenint. A la seva vegada, també farem gràfics de tal forma que puguem comprovar si els nostres models estan millorant o en cas contrari veure que és el que esta fallant tant en el que es relaciona amb la eficiència com amb la precisió dels resultats. Finalment, també ens servira com a mètode de validació les reunions amb el tutor del projecte ja que serviran per exposar els continguts que hem avançat i com ho hem fet i de la mateixa manera seran una forma de veure si el projecte s'esta enfocant correctament.

5 Planificació temporal

Per tal de dur a terme una bona planificació temporal del nostre projecte primerament haurem de fer un càlcul aproximat de quantes hores se li ha de dedicar aquest segons el que esta estipulat a la normativa acadèmica de la FIB. El TFG correspon al que serien 18 crèdits, cada crèdit equival aproximadament a 30 hores, per tant el total de càrrega de feina del TFG hauria de ser d'unes 540 hores. El nostre objectiu a aquesta secció és desglossar aquestes hores en les diferents tasques del nostre projecte a l'hora que aquestes queden distribuïdes dintre del període de temps en el que es compren el TFG.

Segons el calendari acadèmic de la FIB el període de realització compren des del 18 de setembre, amb la impartició de l'assignatura de GEP, i finalitza a la setmana del 22 de gener, amb la exposició final d'aquest. Al tenir aproximadament unes 12 setmanes per dur a terme el projecte hi dedicarem unes 45 hores setmanals de mitja.

A més, en aquesta secció també s'haurà de tenir en compte els riscos els quals ens podem trobar, com solucionar-los, i quins recursos necessitarem durant el projecte.

5.1 Desglossament de les tasques

5.1.1 Gestió del projecte (GP)

El primer que hem de tenir en compte es la gestió del projecte. Per tal de facilitar-nos aquesta part, des de la FIB, s'imparteix l'assignatura de GEP. Aquesta assignatura, que forma part del que és el projecte en si, consta de 3 crèdits equivalents a 90 hores de feina del treball que ens serviran per tal d'organitzar-nos.

Dintre d'aquesta planificació inicial hem de destacar: la definició de l'abast, la planificació temporal, el càlcul del pressupost i l'anàlisi de sostenibilitat del projecte. Les hores de dedicació d'aquestes tasques, tot i que no iguals, estan repartides bastant similar ja que la quantitat de feina també es similar (les tasques GP4 i GP3 es realitzen a la mateixa setmana així que les seves hores estan repartides).

Dintre de la gestió del projecte també comprendrem la documentació com a tal. Aquesta es realitzara simultàniament amb totes les altres tasques durant tot el projecte i se li dedicara una gran part del temps.

Per tant, considerarem les tasques següents:

- Abast **[GP1]** , 35 hores.
- Planificació **[GP2]**, 27.5 hores.

- Pressupost [**GP3**] , 15 hores.
- Sostenibilitat[**GP4**], 12.5 hores.
- Documentació [**GP 5**], 80 hores.

5.1.2 Investigació (INV)

Abans de començar a treballar amb el conjunt de dades i amb diferents tècniques especialitzades per a aquest és important fer una investigació prèvia sobre aquest mètodes i el seu funcionament. Específicament haurem d'entendre i familiaritzar-nos amb els algoritmes de *Zero Shot Learning*, *Few Shot Learning* i els LLM. Per això, dedicarem una mica més d'una setmana per tal d'estar preparats abans de començar a implementar els models.

Concretament considerem les tasques següents:

- Investigació sobre *Zero Shot Learning* i *Few Shot Learning* i la seva utilització per tal de classificar medicaments [**INV1**], 20 hores.
- Investigació sobre els *Large Language Models* i la seva utilització per tal de classificar medicaments [**INV2**], 10 hores.
- Investigació addicional de tècniques i mètodes extres [**INV3**], 10 hores.
- Investigació sobre els conjunt de dades a utilitzar [**INV4**], 5 hores.

5.1.3 Implementació (IMP)

El punt més important del projecte és el de la implementació de les tècniques *Zero Shot Learning* i *Few Shot Learning*. A més, per tal de poder començar a implementar-les, necessitarem que el conjunt de dades hagi estat prèviament pre-processat.

Concretament les taques d'implementació son:

- preprocessament de les dades [**IMP1**], 35 hores per tal de tenir un conjunt de dades net i preparat per als models que utilitzarem.
- Implementació de *Zero Shot Learning* i *Few Shot Learning* per tal de predir quins medicaments son individualment, a quin grup pertanyen, de quina marca i si son o no son medicaments aptes per a ús humà[**IMP2**], 100 hores per assegurar-nos que aquesta tasca e realitza correctament i sense errors ja que tota la experimentació depèn d'ella.
- Neteja del codi implementat, testeig i optimització [**IMP3**], 25 hores on ens assegurarem que el codi es completament llegible i eficient.

5.1.4 Experimentació (EXP)

En aquesta secció definirem aquelles tasques relacionades amb la experimentació. Aquestes tasques pode tenir un temps variable depenent de quan de temps hi vulguem invertir ja que la investigació en si podria no parar fins que nosaltres no ho decidíssim.

- Experimentació amb *Zero Shot Learning*[**EXP1**], 30 hores.
- Experimentació amb *Few Shot Learning* [**EXP2**], 30 hores.
- Creació de test per a l'aplicació dels models [**EXP3**], 30 hores.
- Analitzar els resultats dels experiments i crear una conclusió a partir d'aquests [**EXP4**], 30 hores.

5.1.5 Reunions de seguiment (R)

Les reunions amb el tutor del projecte també s'han de tenir en compte. En concret realitzarem les següents reunions:

- Reunió inicial. És la reunió de principi de curs per tal de comentar amb el tutor de que tractara exactament el nostre projecte i marcarem uns primers objectius i sub-objectius juntament amb unes competències que haurem d'adquirir durant el temps que duri el treball [**R1**].
- Reunions setmanals. Aquestes reunions es realitzaran setmanalment seguint el que es faria amb la metodologia *Agile* per tal d'anar validant el seguiment i corregint petits errors en el transcurs del treball. [**R2**].
- Reunió a mig termini. Reunió que es realitzara a mitat del desenvolupament del treball per validar que s'estan complint les expectatives d'aquest i que és viable continuar amb el objectiu inicial. En cas de que es veies que ens allunyem molt dels terminis previstos tocara reconsiderar el treball i trobar solucions[**R3**].
- Reunió final de projecte. Reunió que realitzarem un parell de setmanes abans d'entregar el projecte per tal de validar resultats, veure que els objectius del projecte s'han complert i corregir errades i possible falta d'informació de la documentació [**R4**].

Totes les reunions seran d'una hora aproximadament menys les reunió inicial, la de mig termini i la final que seran de dos hores per tal de poder discutir correctament el desenvolupament del treball.

5.1.6 Exposició final (EF)

Finalment, preparar la exposició final del treball [**EF1**] amb totes les dades recopilades durant el projecte i defensar-lo davant del tribunal. Hi dedicarem les ultimes 27 hores del projecte ja que consistira bàsicament en preparar una presentació i practica la oratorià sobre el projecte.

5.2 Estimació temporal i dependències entre tasques

En aquesta secció adjuntem la Taula 1 amb les tasques descrites anteriorment i la estimació en hores de cada una d'elles. Les hores han estat mesurades utilitzant el total d'hores que tenim previst pel treball, segons el nivell d'importància de la tasca, més important més hores, i les mètriques descrites anteriorment a cada tasca.

Tasca	**Temps estimat en hores**	**Dependències**
Abast [GP1]	35	R1
Planificació [GP2]	27.5	GP1
Pressupost [GP3]	15	GP2
Sostenibilitat [GP4]	12.5	GP2
Documentació [GP5]	80	R1
Investigació sobre *Zero Shot Learning* i *Few Shot Learning* [INV1]	20	GP3,GP4
Investigació sobre *Large language model* [INV2]	10	GP3,GP4
Investigació addicional [INV3]	10	GP3,GP4
Investigació conjunt de dades [INV4]	5	GP3,GP4
preprocessament dades [IMP1]	35	INV
Implementació de *Zero Shot Learning* i *Few Shot Learning* [IMP2]	100	IMP1
Neteja de codi i optimització [IMP3]	25	IMP2
Experimentació amb *Zero Shot Learning* [EXP1]	30	IMP3
Experimentació amb *Few Shot Learning* [EXP2]	30	IMP3
Creació de tests [EXP3]	30	IMP3
Analitzar resultats [EXP4]	30	EXP1,EXP2,EXP3
Reunió inicial [R1]	2	
Reunions setmanals [R2]	12	R1
Reunió a mig termini [R3]	2	R2
Reunió final de projecte [R4]	2	R3,EXP4
Exposició final [EF1]	27	GP5,R4
Total	**540**	[GP1-GP5], [INV1-INV4], [IMP1-IMP3], [EXP1-EXP4], [R1-R4],[EF1]

Taula 1: Temps estimat en hores i dependències de les tasques

També adjuntem el gràfic de la Figura 4 amb les dependències entre tasques ja que és molt més fàcil de visualitzar. Hem tingut en compte la reunió inicial com a primera tasca ja que és on es van discutir els objectius del treball i a partir d'aquí apareix l'arbre de tasques. A continuació, segueixen les tasques de gestió de projecte, ja que ens serveixen per planificar l'estimació horària del que seguira, i a continuació investigació, implementació i experimentació. A la vegada que tot això s'anirà creant la memòria, i per tant aquesta només és dependent de la reunió inicial, i s'aniran mantinguen les reunions setmanals, que es faran paral·lelament al desenvolupament del projecte. Finalment, tenim la reunió final i la exposició.

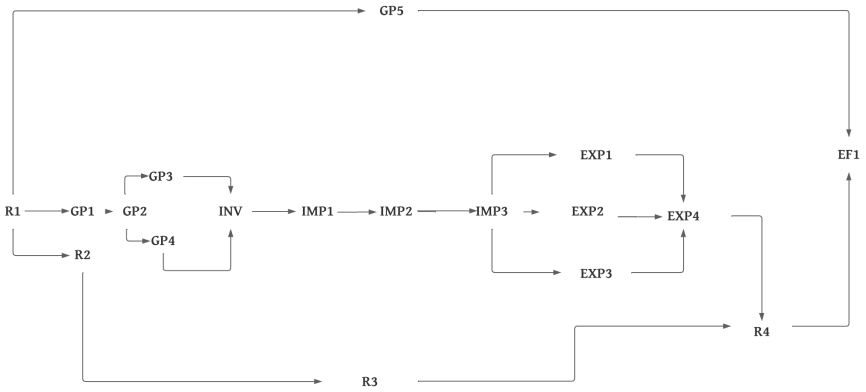

Figura 4: Diagrama de dependències entre tasques

Finalment, també adjuntem un diagrama de Gantt, Figura 5 i Figura 6, que fa referència a la seqüencia de desenvolupament del projecte tinguen en compte les hores per a cada tasca, el període de temps en el qual hauria de ser executada i les dependències entre tasques.

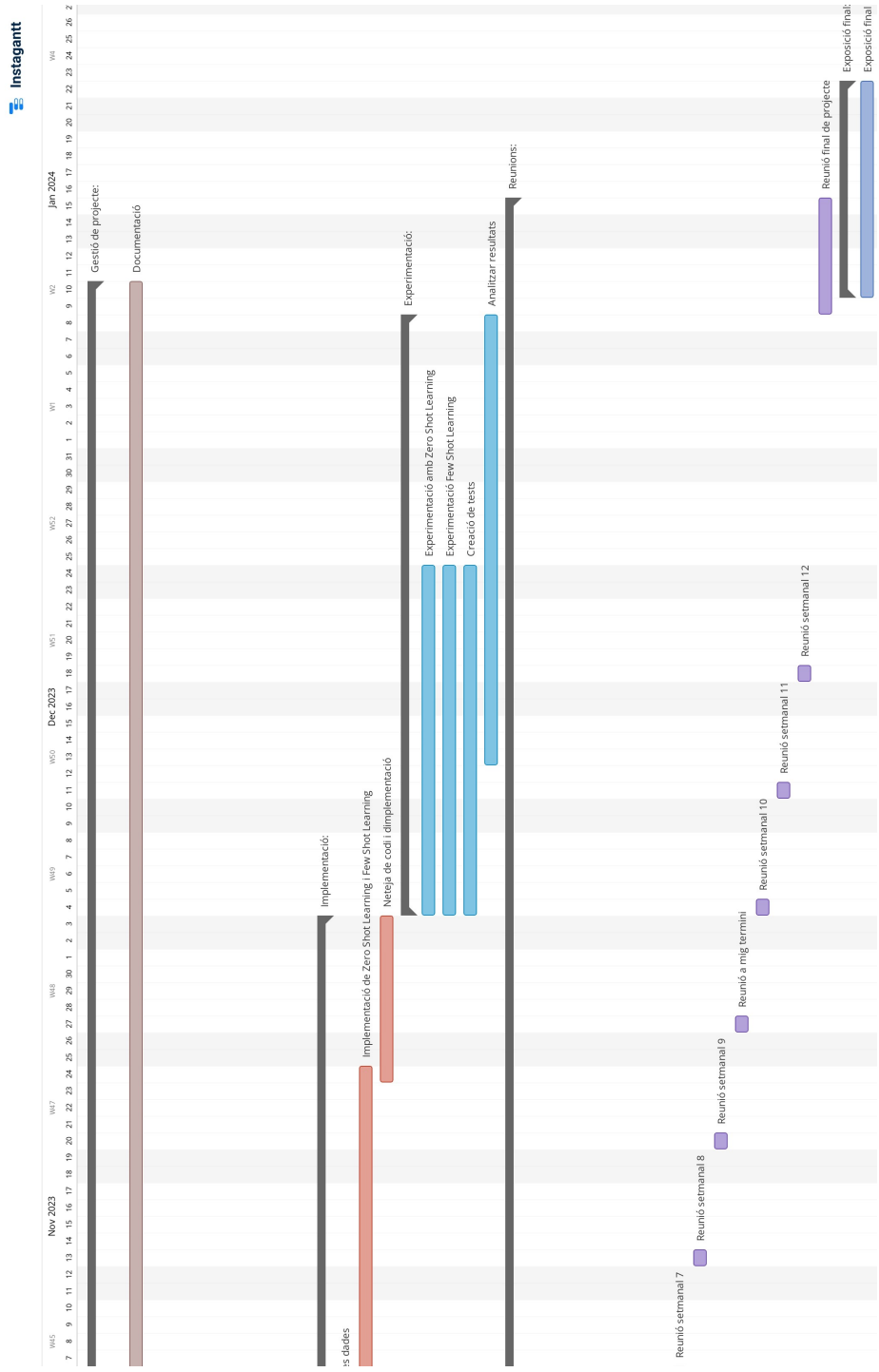

Figura 5: Diagrama de Gantt part 1

	ACTIVITIES	ASSIGNEE	EH	START	DUE	%
	Gestió de projecte:		170h	18/Sep	10/Jan	0%
1	Abast	Xènia Calsalvo	35h	18/Sep	24/Sep	0%
2	Documentació	Xènia Calsalvo	86h	18/Sep	10/Jan	0%
3	Planificació	Xènia Calsalvo	27.5h	25/Sep	02/Oct	0%
4	Pressupost	Xènia Calsalvo	15h	03/Oct	06/Oct	0%
5	Sostenibilitat	Xènia Calsalvo	12.5h	07/Oct	08/Oct	0%
	Investigació:		45h	16/Oct	24/Oct	0%
7	Investigació sobre Zero Sho...	Xènia Calsalvo	20h	16/Oct	18/Oct	0%
8	Investigació sobre Large lan...	Xènia Calsalvo	10h	19/Oct	20/Oct	0%
9	Investigació conjunt de dades	Xènia Calsalvo	5h	21/Oct	21/Oct	0%
10	Investigació addicional	Xènia Calsalvo	10h	23/Oct	24/Oct	0%
	Implementació:		160h	24/Oct	03/Dec	0%
12	Pre-processament de les da...	Xènia Calsalvo	35h	24/Oct	30/Oct	0%
13	Implementació de Zero Sho...	Xènia Calsalvo	100h	31/Oct	24/Nov	0%
14	Neteja de codi i dimplement...	Xènia Calsalvo	25h	24/Nov	03/Dec	0%
	Experimentació:		120h	04/Dec	08/Jan	0%
16	Experimentació amb Zero S...	Xènia Calsalvo	30h	04/Dec	24/Dec	0%
17	Experimentació Few Shot Le...	Xènia Calsalvo	30h	04/Dec	24/Dec	0%
18	Creació de tests	Xènia Calsalvo	30h	04/Dec	24/Dec	0%
19	Analitzar resultats	Xènia Calsalvo	30h	13/Dec	08/Jan	0%
	Reunions:		18h	14/Sep	15/Jan	0%
21	Reunió inicial	Xènia Calsalvo	2h	14/Sep	14/Sep	0%
22	Reunió setmanal 1	Xènia Calsalvo	1h	28/Sep	28/Sep	0%
23	Reunió setmanal 2	Xènia Calsalvo	1h	02/Oct	02/Oct	0%
24	Reunió setmanal 3	Xènia Calsalvo	1h	09/Oct	09/Oct	0%
25	Reunió setmanal 4	Xènia Calsalvo	1h	16/Oct	16/Oct	0%
26	Reunió setmanal 5	Xènia Calsalvo	1h	23/Oct	23/Oct	0%
27	Reunió setmanal 6	Xènia Calsalvo	1h	30/Oct	30/Oct	0%
28	Reunió setmanal 7	Xènia Calsalvo	1h	06/Nov	06/Nov	0%
29	Reunió setmanal 8	Xènia Calsalvo	1h	13/Nov	13/Nov	0%
30	Reunió setmanal 9	Xènia Calsalvo	1h	20/Nov	20/Nov	0%
31	Reunió a mig termini	Xènia Calsalvo	2h	27/Nov	27/Nov	0%
32	Reunió setmanal 10	Xènia Calsalvo	1h	04/Dec	04/Dec	0%
33	Reunió setmanal 11	Xènia Calsalvo	1h	11/Dec	11/Dec	0%
34	Reunió setmanal 12	Xènia Calsalvo	1h	18/Dec	18/Dec	0%
35	Reunió final de projecte	Xènia Calsalvo	2h	09/Jan	15/Jan	0%
	Exposició final:		27h	10/Jan	22/Jan	0%
37	Exposició final	Xènia Calsalvo	27h	10/Jan	22/Jan	0%

Figura 6: Diagrama de Gantt part 2

Primerament, podem veure que la gestió del projecte, exceptuant la documentació, fa referència a les 3 primeres setmanes del projecte. Això es degut a que aquests tempos ja han estan donats per la FIB i fan referència al temps d'entrega dels documents del modul de GEP. Per un altre banda, creiem convenient marcar la documentació durant gaire bé tot el projecte menys a la última part ja que aquell temps és el reservat per corregir errors i preparar la exposició final.

Com ja hem dit abans, les últimes dos setmanes les hem destinat a la preparació de la exposició final. Això és, perquè encara que no se li hagi de dedicar tantes hores en aquesta, teòricament la memòria del TFG ja hauria d'estar acabada en aquell període i només s'haurien de fer els canvis que resultessin oportuns per al tutor.

Per a la resta de tasques, el que hem fet ha estat agafar les hores teòriques restants que quedaven i dividir-les entre les setmanes que comprenien enmig d'aquests dos blocs de tasques. El resultat ha estat d'unes 35 hores per setmana, més les hores que s'hauran d'anar dedicant a la escriptura de la memòria. Igualment, la nostre repartició d'hores tampoc és exacte del tot ja que per exemple creiem oportú reservar algunes hores en la setmana d'exàmens de novembre per tal d'estudiar per altres assignatures i a canvi fer un parell de setmanes unes hores extra. També hem tingut en compte que hi ha algunes feines que no son desglossables i per tant mai podrà ser perfectament exacte. De la mateixa manera, també s'ha de tenir en compte que hi ha dies en els quals tenim pics de productivitat en els que podem treballar fins a 12 hores i n'hi ha d'altres en que amb 4 ja n'hi haurà prou.

Finalment, també tindrem en compte que les setmanes en les quals es celebren festivitats com el Nadal també es veuran perjudicades en el nombre d'hores les quals podrem dedicar al treball i que això ha de ser un visible en el diagrama de Gantt.

Tots els períodes de temps en els que es realitzen les tasques estan basats en el temps assignat per a cada tasca juntament amb el temps que volem invertir en cada setmana per mantenir la coherència.

5.3 Recursos

En tot projecte es necessita un seguit de recursos bàsics per tal de poder dur-lo a terme. Principalment, aquests recursos es separen entre humans, persones implicades en el treball que el duran a terme, i materials, en el nostre cas tant de *software* com de *hardware*. La finalitat d'aquesta secció és descriure quins seran aquests i per a que ens seran útils.

5.3.1 Recursos humans

A nivell de recursos humans necessitarem: un cap de projecte que s'encarregui de les feines de gestió, un enginyer informàtic que resolgui els problemes tècnics del projecte com la programació i el tutor del projecte que s'encarregara de validar els avenços que anem obtenint.

5.3.2 Recursos materials

Com a recursos materials de *hardware* utilitzarem un ordinador de sobretaula amb un processador AMD ryzen 5, una GPU Nvidia GTX 3060 amb 8GB de RAM i el sistema operatiu Windows 10. En quant a *software* farem ús de *LaTex* per a la documentació, el llenguatge *Python* per a les eines d'aprenentatge automàtic, la plataforma de *Google Collab* i el cluster de la facultat per a la implementació i experimentació del projecte, *Trello* i *Instagantt* per a la gestió de les tasques, i finalment també utilitzarem el programari *Sublime* i *VisualCode* per a programació.

5.4 Gestió del risc

Per a un projecte de certa durada com aquest és important especificar com solucionaríem els possibles riscs que ens poden aparèixer mentre el duem a terme.

El primer risc que tindrem en compte és la gestió del temps. Ja que el més important del treball és la implementació i experimentació dels algoritmes de *Zero Shot Learning* i *Few Shot Learning* se'ls ha sobreestimat i donat més temps del que teòricament seria necessari, probablement amb 70-80 hores seria suficient, per tal de realitzar les seves tasques. D'aquesta manera ens assegurem del compliment del objectiu principal del treball.

En el cas d'un problema greu del tipus accident o malaltia, depenent de la seva gravetat, reduiríem els objectius alternatius com experimentar amb més LLM i reduir el nombre de conjunts de dades amb els que testejar per tal de poder seguir complint amb els objectius principals. També es podrien utilitzar hores extres, en les setmanes on les hem reduït, com la setmana d'exàmens o les setmanes de festivitats, on actualment hi dediquem unes 10 hores en cada una, i augmentar aquestes hores a unes 35, com a les altres setmanes, per tal de fer front al incident.

Un altre problema que enfrontem, ja que treballem des d'un despatx alie a casa nostre, és la caiguda de la xarxa. En aquest cas, depenent de la llargada de la situació, recorreríem a utilitzar un portàtil, el que es tracta d'un recurs extra, i moure'ns a alguna biblioteca pública on poder seguir avançant amb el treball.

En quants als petits errors s'utilitzaran les reunions setmanals amb el tutor per

tal de ser detectats i re-encaminats a temps. Això forma part de la metodologia *Agile* que utilitzem en el treball.

6 Gestió econòmica

En aquesta secció tractarem el tema econòmic i farem un estudi per tal d'estimar els costos associats al nostre projecte. Els costos estaran categoritzats segons el seu tipus ja siguin de personal, genèrics, de contingència o imprevists.

6.1 Pressupost

6.1.1 Costs de personal

A partir de la planificació de les tasques identificarem els perfils del personal, veieu Taula 2, que intervindran en aquest projecte: cap de projecte, investigador, programador i tester. El cost per hora en brut de cada perfil s'ha calculat a partir del sou anual de cada rol, dividit entre les hores legals anuals (unes 1820). Per calcular el cost per hora amb la seguretat social utilitzem el càlcul previ i multipliquem per 1.28 ja que utilitzem el percentatge de cotització del 2022 que correspon al 28%.

Rol	**Cost per hora en brut**	**Cost per hora amb seguretat social**
Cap de projecte	28.83€/h	37.48€/h
Investigador	17.96€/h	23.35€/h
Programador	17.72€/h	23.04€/h

Taula 2: Cost per hora dels diferents rols. Elaboració pròpia amb informació extreta de *Glassdoor*.

A continuació, a la Taula 3, veiem els costos concrets de cada tasca en funció del temps en hores que s'ha tardat en realitzar-la i els rols involucrats en aquestes. També adjuntem els costs totals de cada grup de tasques i el cost total del projecte final. En tot moment es suposa que son treballadors a temps complert i amb salari indefinit per tal de no complicar encara més els càlculs.

Tasca	Temps	Rols	Cost Net
Gestió de projecte [GP]	**169.5h**	**-**	**6337.91€**
Abast [GP1]	35h	CP	1009.05€
Planificació [GP2]	27.5h	CP	792.83€
Pressupost [GP3]	15h	CP	432.45€
Sostenibilitat [GP4]	12.5h	CP	360.38€
Documentació [GP5]	80h	CP, I	3743.2€
Investigació [INV]	**45h**	**-**	**808.2€**
Investigació sobre *Zero Shot Learning* i *Few Shot Learning* [INV1]	20h	I	359.2€
Investigació sobre *Large language model* [INV2]	10h	I	179.6€
Investigació addicional [INV3]	10h	I	179.6€
Investigació conjunt de dades [INV4]	5h	I	89.8€
Implementació [IMP]	**160h**	**-**	**2835.2€**
preprocessament dades [IMP1]	35h	P	620.2€
Implementació de *Zero Shot Learning* i *Few Shot Learning* [IMP2]	100h	P	1772€
Neteja de codi i optimització [IMP3]	25h	P	443€
Experimentació [EXP]	**120h**	**-**	**2126.4€**
Experimentació amb *Zero Shot Learning* [EXP1]	30h	P	531.6€
Experimentació amb *Few Shot Learning* [EXP2]	30h	P	531.6€
Creació de tests [EXP3]	30h	P	531.6€
Analitzar resultats [EXP4]	30h	CP, P	531.6€
Reunions [R]	**18h**	**-**	**1161.18€**
Reunió inicial [R1]	2h	CP, I, P	129.02€
Reunions setmanals [R2]	12h	CP, I, P	774.12€
Reunió a mig termini [R3]	2h	CP, I, P	129.02€
Reunió final de projecte [R4]	2h	CP, I, P	129.02€
Exposició final[EF]	**27h**	**-**	**778.41€**
Exposició final [EF1]	27h	CP	778.41€
Total	**530h**	**-**	**14047.3€**

Taula 3: Relació entre les tasques, les hores, el rol encarregat de dur-les a terme (CP - cap de projecte, P - programador, I - investigador) i el cost total d'aquestes.

6.1.2 Costs genèrics

A part dels costs de personal és important tenir en compte un seguit de costs genèrics que suposen una gran part del pressupost total del projecte. A continuació parlem dels més importants:

1. **Espai de treball.** És on durem a terme tot el projecte. Ja que el projecte el realitzarem des del despatx d'una casa calcularem el cost aproximat de l'habitació ocupada. L'habitació en concret té un 21 metres quadrats. Actualment, alquilar un metre quadrat té un cost d'11.34€[7] al mes i la durada del nostre projecte és d'uns 5 mesos (considerarem 5 ja que encara que el cinquè sigui incomplet no es pot alquilar per menys d'un mes) per

tant el cost total del espai és de 21x11.34x5 = **1190.7€.**

2. **Amortitzacions.** L'únic material a amortitzar serà l'ordinador de sobretaula que s'utilitzara per al projecte el qual té un cost total d'uns 1500€. Ja que suposarem que de mitjana solen tenir una vida útil de 60 mesos (5 anys) l'amortització del recurs és de 5/60 x 1500 = **125€.**

3. **Consum elèctric.** El preu de la llum varia segons la hora en la qual la utilitzem i per tant agafarem un valor de referència de les hores en les que solem dedicar més temps al projecte, és a dir, les 11 a.m. A aquesta hora el preu de la llum esta en 0.19512 €/kWh [8], preu que utilitzarem per tal de fer una estimació del funcionament del ordinador de sobretaula que utilitzarem. Un ordinador de sobretaula té un cost d'uns 180W de mitja i com l'utilitzarem les 540 hores que tenim previstes per al treball tenim un total de 0.19512 x 0.180 x 540 = **18.97€.**

4. **Factura d'internet.** La factura d'internet del despatx en el qual treballem ascendeix a 60€ mensuals. Com el treball que realitzarem necessita d'unes 4 hores diàries durant 5 mesos, la part proporcional de la factura d'internet és de 5 x 60 x 4/24 = **50€.**

Concepte	**Cost**
Espai de treball	1190.7€
Amortitzacions	125€
Consum elèctric	18.97€
Factura d'internet	50€
Total CG	**1384.67€**

Taula 4: Cost per hora dels diferents rols. Elaboració pròpia amb informació extreta de *Glassdoor*.

Podem observar un resum dels costs i el total de la suma a la Taula 4.

6.1.3 Contingències

Com en tot projecte, és important contemplar la possibilitat que durant la realització d'aquest es poden donar imprevistos que impliquin un contratemps o/i complicacions durant el projecte. Ja que el temps és directament proporcional a la part del pressupost del personal sempre s'ha de tenir un pla de contingència per aquestes situacions poder fer front la situació. També s'ha de tenir en compte un augment en els costs genèrics ja que aquest temps extra també influirà en quanta factures, lloguers, depreciacions, etc. Per tot això agafarem un valor de contingència del 15%, en la mitja del que afronta un projecte de *software*.

El càlcul final resulta de la següent manera: (TOTAL CPA + TOTAL CFG) x 0.15 = (14047.3€ + 1384.67€) x 0.15 = **2314.80€** de cost de contingència.

6.1.4 Imprevists

Per finalitzar, es presenta el càlcul dels diferents imprevists i obstacles que ens poden passar durant la realització del projecte. Aquests imprevists son els que ja han estat comentats durant la planificació temporal i dels quals quantificarem en diners i probabilitat de que succeeixin a continuació.

Imprevist	**Despesa**	**Probabilitat**	**Cost**
Nou ordinador portàtil	500€	10%	50€
Increment del temps d'implementació dels algoritmes(20h)	354.4€	20%	70.88€
Increment del temps d'experimentació(20h)	354.4€	40%	141.76€
Total imprevists			**262.64€**

Taula 5: Relació imprevists amb el cost d'aquests, probabilitat de que succeeixi i cost final d'aquest.

A a la Taula 5 queden representades les probabilitats de cada un d'aquests. Hem assignat un 10% a l'haver de comprar un nou ordinador portàtil ja que l'actual ordinador de sobretaula que utilitzem esta en el seu segon any de vida i no presenta problemes així que veiem poc probable que deixi de funcionar. Per un altre banda, hem assignat un 20% a necessitar més temps per a la implementació d'algoritmes ja que, encara que poc probable, per cas de malaltia o similar els podríem necessitar. També hem assignat una probabilitat a l'increment del temps d'experimentació d'un 40% ja que la experimentació sempre es pot allargar més i crear nous casos amb els que provar així que és més probable de que es dones aquest cas.

6.1.5 Cost total

A la Taula 6 es mostren tots els costs, prèviament descrits, desglossats segons el tipus i el total del cost del projecte.

Concepte	**Cost**
Costs de personal	14047.3€
Costs genèrics	1384.67€
Contingències	2314.80€
Imprevists	262.64€
Total	**18009.41€**

Taula 6: Resum dels costs de cada categoria i el total.

6.2 Control de gestió

Un cop definit el pressupost inicial hem de definir com es dura a terme el control de gestió d'aquest. El control de gestió ha de ser quantificable, és a dir, necessita d'indicadors numèrics per tal de tenir un correcte funcionament. Utilitzarem les reunions setmanals per veure si s'han produït desviacions respecte al pressupost inicial. Hem decidit fer-ho durant aquestes reunions ja que així es produiran de manera periòdica mentre parlem amb el client, en aquest cas el tutor del TFG, i es podrà comparar correctament amb les hores estimades de feina.

Per tal de controlar els imprevists al finalitzar les tasques també apuntarem el cost que aquestes han tingut i ho compararem amb les previsions d'imprevistos i contingència. Seguint tots aquests passos serà fàcilment comprovable si hem d'acurtar alguna tasca o veure si hem de modificar el pressupost.

Les mètriques que utilitzarem seran les següents:

- **Desviació de cost =** (Cost Estimat - Cost Real) x Consum d'Hores Real
- **Desviació de consum =** (Consum d'Hores Estimat - Consum d'Hores Real) x Cost Estimat
- **Desviació d'hores =** Hores Estimades - Hores Reals
- **Desviació cost d'imprevists =** Cost Estimat d'Imprevists - Cost Real d'Imprevists

7 Sostenibilitat

7.1 Autoavaluació

Després de realitzar l'enquesta de sostenibilitat he vist que, tot i que els coneixement amb la matèria de sostenibilitat no son baixos, a l'hora de tenir que posar-los en pràctica i tenir en compte tots els factor implicats no tinc per costum el plantejar-los directament en els meus projectes.

Des del meu punt de vista, tot i que en un projecte sempre es té en compte la dimensió econòmica d'aquest, poques vegades parem a reflexionar sobre la dimensió ambiental i social. Si que és veritat que les coses bàsiques relacionades amb l'àmbit ambiental com per exemple intentar utilitzar el mínim nombre de recursos possibles o intentar reutilitzar tot el material possible es solen duu a terme. Peró per un altre banda, també crec que això només succeeix perquè en certa forma interessa des de la part econòmica.

També crec destacar que la part social és la que es veu més apartada en els projectes d'enginyeria, ja que d'alguna manera queda com a l'oblit el fet de que la repercussió social pugui ser més important respecte a els guanys financers o els avenços científics. Això es veu clarament en el reflex del camp del meu TFG, la IA, on últimament s'esta fent cada cop més èmfasis amb el nombre de feines que poden ser substituïdes per aquesta i com amenaça a la nostre forma de societat actual.

En conclusió, crec que tots els projecte haurien de tenir en compte aquests 3 àmbits i intentar d'alguna manera quantificar-los sempre ja que em sembla la millor forma d'assegurar la sostenibilitat i viabilitat d'aquest mateix. En el cas del meu projecte, crec que la utilització de l'IA amb finalitats mèdiques és clarament avantatjosa per a la societat i s'intentara tenir en compte tot el que em mencionat anteriorment per tal de no nomes dur a terme el projecte amb èxit sinó que sigui clarament d'ajuda d'ara en endavant.

7.2 Dimensió econòmica

Respecte a la dimensió econòmica de la matriu de sostenibilitat, tot i que el cost del projecte pot semblar elevat s'ha de se tenir en compte que gairebé tots els recursos s'han destinat a personal cosa que difícilment es pot reduir i no es malgasten diners en recursos innecessaris. Tinguen en compte els beneficis que pot aportar el projecte i els costs en personal sanitari que es retallarien a la llarga crec que el pressupost es adient per al projecte.

Actualment, els costs derivats de la solució que proposem es resolen amb temps del personal sanitari que es podria invertir en altres tasques més adient per ells. Si es veritat que també hi ha altres projectes amb IA que poden ser utilitzats amb la mateixa finalitat que el nostre però que encara no s'utilitzen suficient-

ment per a ser comparables econòmicament amb aquest.

A més, tot i que s'hagi de fer la despesa inicial del projecte, un cop creat aquest el manteniment és molt baix i permetrà una clara millora econòmica al poder prescindir de personal sanitari que gasti les seves hores d'horari laboral en aquest tipus de tasques.

7.3 Dimensió ambiental

Aquest projecte no necessita de nous recursos, en quant a maquinaria es refereix, si no que es serveix de recursos prèviament ja existents com poden ser l'ordinador de sobretaula de l'autor, el qual utilitza també per a altres tasques, o el clúster de la universitat, el qual també té moltes altres finalitats. Aquest fet implica que des del principi no es planteges massa la dimensió ambiental en el projecte ja que com no es necessitava de nous recursos tampoc suposava un gran canvi cap al medi ambient.

Tot i així, un cop que ens hem endinsat més en el treball ens hem adonat de que tot i que no necessitem de més material per a dur a terme el projecte, si que hauríem de parlar del tema de la petjada de carboni que deixen els LLMs. Com estem parlant de models massius de llenguatge, els quals necessiten d'una gran quantitat de recursos computacionals, a la seva vegada estem parlant d'un ús d'energia directament proporcional als recursos el qual podria ser molt elevat. Les causes del augment d'electricitat necessaris pel projecte van des dels requisits de potència i *hardware* amb els que ens trobem, les GPU son les principals causants d'això, a el ús de gran quantitats de dades d'entrenament les quals estan emmagatzemades a el nuvol, això també ens faria tenir en compte la maquinaria i electricitat del servidor que les conté, o simplement a la necessitat de fer un ús continu dels càlculs tant com per a *fine-tuning* com per a les prediccions. Tot i que per abordar aquest seguit de problemes s'estan duent a terme diverses investigacions per millorar la eficiència energètica i per fomentar pràctiques més sostenibles, esta clar que avui en dia això segueix siguen una problemàtica vigent la qual hem de tenir en compte i ser conscients a l'hora de fer ús d'aquestes tècniques i models.

7.4 Dimensió social

Personalment, a mesura que he anat endinsant-me en el mon de la informàtica i tocant diferents aspectes d'aquesta m'ha interessat especialment el mon de la IA. En els últims anys sobretot vaig tractar de triar el màxim d'assignatures relacionades amb aquest camp i finalment vaig decidir basar el meu projecte en el reconeixement de textos. A partir d'aquí, crec que la realització d'aquest treball em permetrà tant profunditzar en el meus coneixement en l'aprenentatge automàtic coma la seva vegada veure aplicacions realment practiques que permeten una millora real en el mon.

La nostre solució millorara directament la cerca d'informació important a textos científics, mèdics i hospitalaris estalviant així una quantia de temps important a el personal relacionat amb el sector. Per altre banda, també permetrà tenir més gestionat els costos dels medicaments fent així que el pressupost sobrant pugui anar destinat a altres projectes socials. Finalment, també és important la millora social en quant a la vida d'alguns pacients que podrien preveure els efectes secundaris dels medicaments abans de tenir que ingerir-los i així buscar alternatives millors.

Tot i que no es podria dir que el projecte sigui una necessitat real en el sentit de que actualment no hi hagi cap forma de solucionar el problema, si és cert que d'aquesta manera millorarem molt la solució existent. Creiem que amb el desenvolupament del projecte podem aconseguir retallar els temps de les tasques de la sanitat, temps molt valuós i actualment molt necessari. A mes, al tractar-se d'un projecte relacionat amb la extracció d'informació a partir de texts també es pot extrapolar a molts altres camps i tenir diverses utilitats.

8 Conjunt de dades: *DDI corpus*

8.1 Resum bàsic del conjunt de dades

Per al nostre projecte utilitzarem el *DDI corpus*[9], dels autors, María Herrero-Zazo, Isabel Segura-Bedmar, Paloma Martínez i Thierry Declerck, el qual conte informació sobre substancies farmacològiques i les interaccions entre medicaments (nosaltres ens centrarem en la primera), veieu un resum a la Figura 7. Aquest corpus esta creat a partir del conjunt de dades *Drug Bank*, del qual conte 572 documents descrivint les interaccions entre drogues, i dels *abstracts* de *MEDLINE*, que contenen un total de 142 *abstracts* que tenen com a subjecte principal la interacció entre drogues. El total de substancies farmacològiques tractades és de 18502, i el nombre de DDIs anotats 5028, incloent tant les interaccions farmacocinètiques com les farmacodinàmiques. Aquest, s'ha utilitzat en una competició (DDIExtraction 2013), amb resultats destacats que demostren la seva utilitat per al reconeixement de substàncies farmacològiques i la detecció de DDIs en textos biomèdics.

	DRUGBANK Training			**MedLine Training**		
	Total	**Avg. Doc**	**Avg. Sentences**	**Total**	**Avg. Doc**	**Avg. Sentences**
Documents	572			142		
Sentences	5675			1301		
Drugs	8197			1228		
Brand	1423			14		
Group	3206			193		
No human	103			401		
Total Drugs:	**12929**	**22.6**	**2.3**	**1836**	**12.9**	**1.4**
DDIs						
ddi	178			10		
advice	819			8		
effect	1548			152		
mechanism	1260			8162		
Total DDIs:	**3805**	**6.6**	**0.7**	**232**	**1.6**	**0.2**

Figura 7: Taula resum del conjunt de dades *DDI corpus*[10].

El conjunt de dades esta semànticament anotat i se'ns proporciona en format XML, veieu Figures 8 i 9. L'anotació semàntica és la següent[10]:

- ***Document element.*** És l'arrel del document i només conte un únic atribut **id** el qual esta compost pel nom del corpus (*DDI-DrugBank* o *DD-MedLine*) i un identificador que comença per "d"seguit d'un numero.
- ***Sentence element.*** Cada oració del document té un *Sentence element*. Aquest esta constituït d'un atribut **id**, compost pel nom del corpus, l'id del document, i un id començat per "s"seguit d'un index de la oració (inicia en 0) i també per al **text** de la oració.

- ***Entity element.*** Compost pels següents atributs:
  - **id:** compost pel nom del corpus, l'id del document, l'id de la oració i un id començat per "e"seguit d'un index de l'entitat de la oració.
  - ***charOffsets:*** conte les posicions d'inici i final de la menció a la oració.
  - ***text:*** guarda el text en si.
  - ***type:*** guarda tant el tipus de la substancia farmacològica i si aquesta és una droga, la marca, de quin grup forma part, i si és apte o no per al consum humà
- ***Discontinuos names.*** N'hi ha de dos tipus:
  - **Coordinadors:** la oració conte dos substancies farmacològiques diferents i per tant té una estructura coordinada. Això es veu reflectit en el *charOffset* on es marquen les posicions d'inici i final de cada una de les parts de la menció.
  - **Abreviacions:** algunes vegades utilitzem abreviacions, acrònims o altres noms per tal de fer referencia a les substancies a més del seu nom. Quan utilitzem aquestes, una menció amb una abreviació només conta com una sola entrada.
- ***DDI element.*** Fa referència a les interaccions entres les drogues. Esta compost pels següents atributs:
  - **id:** compost per l'id del corpus, del document, de la oració i un id extra començat per "dï seguit per l'index del ddi a la oració.
  - **e1:** guarda l'id de la primera entitat que interactua.
  - **e2:** guarda l'id de la segona entitat que interactua.
  - **type:** guarda el tipus de la interacció, aquest pot ser un ddi, una advertència, un efecte o un mecanisme.

```
<?xml version="1.0" encoding="UTF-8"?>
- <document id="DDI-MedLine.d11">
  - <sentence id="DDI-MedLine.d11.s0" text="Immunosuppressive drugs and their complications. ">
      <entity id="DDI-MedLine.d11.s0.e0" text="Immunosuppressive drugs" type="group" charOffset="0-22"/>
    </sentence>
    <sentence id="DDI-MedLine.d11.s1" text="Drugs that suppress the immune system are widely used. "/>
    <sentence id="DDI-MedLine.d11.s2" text="They are part of the treatment of patients with organ transplants, malignancy, and increasingly those with conditions such as psoriasis, rheumatoid arthritis, and liver and bowel disease in which inflammation is an aetiological factor. "/>
  - <sentence id="DDI-MedLine.d11.s3" text="Because of the broadening indications for immunosuppressive drugs, and the prolonged survival in conditions for which they are being used, many patients on immunosuppression are now cared for in the community or seen in non-specialist hospitals, usually in close collaboration with a specialist. ">
      <entity id="DDI-MedLine.d11.s3.e0" text="immunosuppressive drugs" type="group" charOffset="42-64"/>
```

Figura 8: Exemple de la semàntica del document.

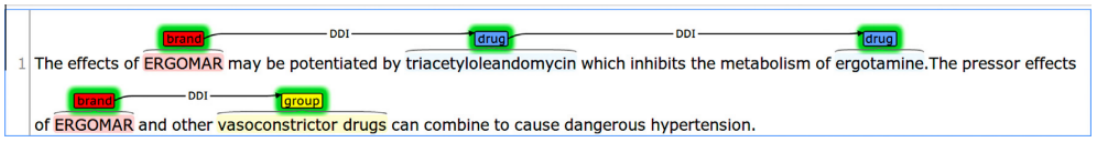

Figura 9: Exemple de DDI: Efecte i mecànisme.

8.2 Decisió sobre el conjunt de dades

El conjunt de dades proporcionat en format XML, amb informació sobre frases, entitats i relacions, ofereix una estructura jeràrquica que facilita el processament i la comprensió. Les entitats, com els medicaments, estan etiquetades amb detalls addicionals, com el tipus i la posició en el text, simplificant així la tasca de reconeixement d'entitats.

Aquest format també inclou informació sobre les relacions entre les entitats, presentada de manera clara mitjançant marques específiques. Cada frase manté el seu context original, el que resulta beneficiós per a tasques de modelatge del llenguatge.

L'estructura del conjunt de dades, juntament amb les relacions prèviament definides, podria ser explotada per a tasques de classificació. A més, l'ús de models de llenguatge avançats i l'aprenentatge amb pocs exemples podrien aprofundir encara més en l'anàlisi de textos relacionats amb la interacció de medicaments, proporcionant coneixements significatius i útils per a aplicacions clíniques.

9 Preprocessament del conjunt de dades

9.1 Objectiu del preprocessament de les dades

El primer a tractar a l'hora de pre-processar el conjunt de dades és triar quin format serà el més adient per als nostres models. Al tractar-se originalment d'un conjunt de dades codificat en XML tenim moltes opcions ja que en qualsevol cas s'havia de fer un tractament de tota la informació. El nostre conjunt de dades ja ve separat entre els tres subconjunts necessaris per al model; test, entrenament i avaluació. Segons quin d'aquests estiguem tractant necessitarem fer un preprocessament més senzill o un de més complex.

En el primer cas, només processarem la informació per obtenir les sentencies del conjunt de dades de la mateixa manera que estan escrites. És a dir, serà un reflex exacte del que conte el text inicial. Aquest processament de la informació ens serà útil per a tenir els conjunts de dades els quals volem classificar, tant el conjunt de test, com el conjunt d'avaluació.

Per un altre banda, necessitarem uns conjunts de dades que tinguin les respostes que esperem predir, en aquests casos els conjunts tindran tant la informació dels anteriors com una informació extra que contindrà d'alguna forma el resultat de les substàncies farmacològiques predites i la classificació del grup que pertanyen. Aquests conjunts de dades seran tant el conjunt d'entrenament, que utilitzarem més endavant per al *fine-tuning* del model, com el conjunt de test, que aquest cop utilitzarem per a realitzar l'avaluació final dels resultats predits.

9.1.1 Objectiu del preprocessament de les dades per a la sentència inicial

La sentencia inicial fa referència a la oració sense canvi algun del text en el qual volem trobar i classificar les substàncies farmacològiques. Per a això necessitem indicar-li al model d'alguna forma que la sentencia inicia, un exemple senzill per fer això és afegint "Sentence: "a l'inici. A continuació d'això simplement voldrem el text extret del extracte de XML. A la Figura 10 i a la Figura 11 podem veure un exemple del format esperat.

```
<sentence id="DDI-MedLine.d27.s1" text="When the dose of picrotoxin is minimized to 0.5 mg/kg such an effect is not
observed. ">
```

Figura 10: Sentencia abans de pre-processar.

```
Sentence:  When the dose of picrotoxin is minimizen to 0.5 mg/kg such an effect is not observed.
```

Figura 11: Sentencia després de pre-processar.

9.1.2 Objectiu del preprocessament de les dades per a la resposta

La resposta fa referència a la classificació de les substancies farmacològiques de la sentencia inicial. Les dos opcions que van agafar més pes per a codificar-ho van ser les següents:

- **Format de llista.** En el format de llista volem que aparegui cada droga llistada amb la classificació a la qual pertany. Per a fer-ho iniciarem afegint una etiqueta on aparegui que la llista comença, com per exemple "Drugs: ", i afegirem a cada línia el tipus de la droga en concret, un guio, i el text que conte en nom de la droga tal i com veiem a la Figura 12.

```
Sentence:  When the dose of picrotoxin is minimized to 0.5 mg/kg such an effect is not observed.
Drugs:
drug_n - picrotoxin
```

Figura 12: Llista de drogues després de pre-processar.

- **Format de marques.** El format de marques és el següent; agafem el text original, busquem les substàncies farmacològiques del text, classifiquem cadascuna d'elles, ho marquem amb un format pseudo-HTML en el mateix text i marquem que es tracta d'una resposta al text afegint un "Response: "a l'inici de la resposta. Podem veure un exemple a la Figura 13.

```
Response:  When the dose of <drug_n>picrotoxin<\drug_n> is minimized to 0.5 mg/kg such an effect is not observed.
```

Figura 13: Resposta amb marques després de pre-processar.

9.2 *XML Parsing*

Com hem vist anteriorment, el nostre conjunt de dades es troba codificat en XML aixi que el primer pas serà el de descodificar-lo. La funció *parse* s'aplica sobre l'arxiu XML i ens permet analitzar-lo gramaticalment obtinguen-te un arbre DOM, veieu Figura 14. Aquest arbre no és res més que una representació de l'arxiu de forma jeràrquica que transforma cada element del document, com etiquetes, atributs i texts, en nodes de l'arbre. Concretament i en el nostre cas, l'arbre arrel representara el document complert, els seus fills representaran les *sentences* i els fills dels fills les *entities*. D'aquesta manera tindrem una forma d'iterar, senzilla i coneguda, sobre les dades.

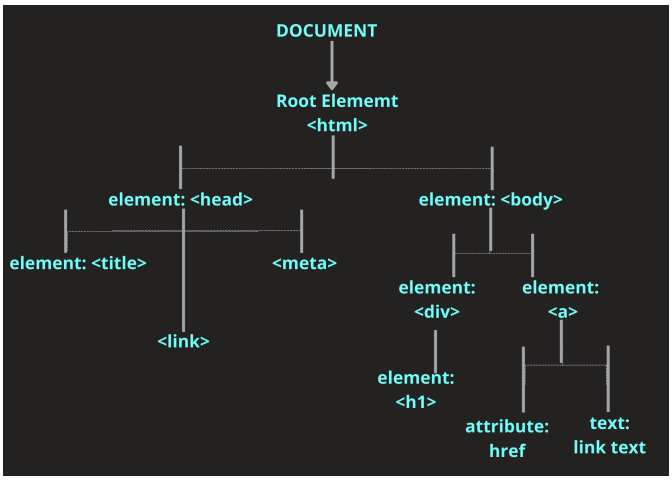

Figura 14: Exemple d'arbre DOM.

9.3 Extracció de la informació

L'objectiu ara consistira en a partir de l'arbre DOM obtingut aconseguir extreure la informació necessària, aquesta dependrà de quin objectiu de preprocessament estiguem duent a terme. Per a fer-ho començarem iterant sobre les *sentences* fent us de la funció *getElementsByTagName* obtinguen-te totes les etiquetes que tenen aquest nom de l'arbre. Dels atributs que conte la *sentence* ens fixarem en el atribut de text i, en el cas de fer el processament que necessita de les substancies farmacològiques, també en les entitats. Per a iterar dintre de les entitats es segueix exactament el mateix procés que per a les *sentences*. Finalment, d'aquestes segones només ens quedarem amb els atributs de text i de tipus ja que son els únics rellevants per a les nostres prediccions.

9.4 Classificació de la informació

Aquest apartat només serà necessari en el cas d'estar elaborant la resposta. En aquest cas, ja tindríem extreta tant la informació que fa referència al text com a les entitats que ens trobem en el text. El procediment d'aquest pas dependrà de si volem les substàncies en el format llista o en el format de marques. En el primer cas, simplement consistira en escriure el tipus de la droga i el seu nom en el format més convenient ja descrit anteriorment. En canvi, en el segon cas substituirem el contingut del text per a generar la resposta. Concretament, buscarem l'aparició del text que fa referencia a la entitat en el text original i marcarem aquesta aparició en el text amb unes marques de pseudo-HTML que a la seva vegada indiciaran el tipus de la substància marcada.

9.5 *Text Cleaning*

Per al preprocessament també hem hagut de fer un petit *Text Cleaning*. En el nostre cas, només ha suposat eliminar espais i salts de línia innecessaris. Entre els salts de línia també hem hagut de tenir en compte les codificacions en XML "
", "
"que ocasionaven un comportament no esperat si no ho suprimíem del text original.

9.6 Decisió final del preprocessament

Finalment, i després d'haver realitzant un seguit de proves hem decidit utilitzar el format de marques i deixar enrere el format de llista per al conjunt de dades a partir d'ara. Aquesta decisió ha estat presa degut als següents factors:

- El primer format conserva el context original del text juntament amb les etiquetes de drogues. Això permet als LLM entendre el context més ampli de les frases, ja que conserva les paraules i la seva disposició.
- El primer format és més coherent amb anotacions detallades i estructurades. Els *Large language models* poden aprofitar aquesta estructura per aprendre patrons i relacions més detallades.
- El primer format proporciona un marc més ric per al *few/zero shot learning* ja que els models poden ser entrenats en contextos més detallats. Això pot millorar la capacitat del model per comprendre i generar textos relacionats amb drogues, fins i tot en situacions on no ha estat explícitament entrenat.

En resum, el primer format és més adequat per a *Large language models* i *few/zero shot learning* perquè conserva el context original, les relacions semàntiques i proporciona una estructura més rica i coherent que pot ser aprofitada pel model durant l'entrenament i la generació de textos.

10 Implementació dels models

10.1 *Overview*

En aquesta secció veurem una visió general de la implementació i l'aplicació dels models per al nostre projecte. L'objectiu és el de comprovar la precisió i efectivitat d'aquests models en el problema concret que anem a tractar. El projecte consisteix en utilitzar diferents estratègies d'aprenentatge en una sèrie de LLM de diferents mides i especificacions per tal de veure les diferències en quant a rendiment i precisió en els resultats.

10.2 *Overview* dels models

10.2.1 GPT-2

GPT-2, o "Generative Pre-trained Transformer 2", és un model desenvolupat per OpenAI, una empresa d'investigació en intel·ligència artificial. Nosaltres utilitzarem la plataforma i les llibreries de Hugging Face, anomenades "transformers", que faciliten l'ús i la implementació de diversos models de llenguatge com aquest[11].

GPT-2 és un model pre-entrenat en un corpus molt extens de dades en anglès, mitjançant un procés auto-supervisat. Això significa que el model va ser pre-entrenat només amb textos sense cap tipus de preprocessament mitjançant un procés automàtic per generar entrades i etiquetes a partir d'aquests textos. Concretament, el model va ser entrenat per tal de poder predir la següent paraula en una oració.

D'una manera més precisa, les entrades són seqüències de text continu d'una certa longitud, i els objectius són la mateixa seqüència, desplaçada un símbol (paraula o fragment de paraula) cap a la dreta. El model utilitza internament un mecanisme de màscara per assegurar-se que les prediccions per al símbol 'i' només utilitzen les entrades de 1 a 'i', però no els símbols futurs.

D'aquesta manera, el model aprèn una representació interna de la llengua anglesa que després es pot utilitzar per extreure característiques útils per a tasques posteriors. No obstant això, el model és més eficient en allò pel qual va ser pre-entrenat, que és la generació de textos a partir d'una indicació.

Hi ha diverses versions amb diferents mides del model; GPT-2 amb 124M de paràmetres, GPT-2 Medium amb 355M, GPT-2 Large amb 774M i GPT-2 XL amb 1.5B.

Forts del model:

- Comprensió del context. GPT-2 captura bé la informació contextual, cosa beneficiosa per a tasques que necessiten la comprensió del significat com

la classificació.

- Aprenentatge de transferència. Al ser un model entrenat en un corpus de dades tant diverses el coneixement es pot aprofitar amb uns ajustament per a tasques de classificació especifiques tot i diferir del seu entrenament original.

- Compatibilitat amb *few-shot Learning* i *zero-shot Learning*. Això significa que, en lloc de requerir grans conjunts de dades d'entrenament etiquetades, el model pot ser personalitzat i utilitzat per tasques específiques amb una petita, o nul·la, quantitat d'exemples, fent-lo més eficient i adaptable a diverses aplicacions.

- Entrades variables. Les seqüències d'entrada del model poden ser de longitud variable, la qual cosa el fa versàtil per a tasques en què la longitud del text d'entrada pot variar.

Debilitats del model:

- Recursos computacionals. Depenent de quina versió del model estem utilitzant és necessitara d'uns recursos computacionals significatius i un temps de computació del qual pot ser que no disposem.

- Sobre parametrització. La gran quantitat de paràmetres és també un problema ja que que si sobre paramètritzem el model per a masses tasques pot tenir costos computacionals innecessaris.

- Risc de sobre ajustament. Ajustar un model gran com és GPT-2 a un conjunt de dades petit comporta un risc de sobre ajustament.

- Informació no filtrada. La informació que s'ha utilitzat per a l'entrenament del model no ha estat filtrada, una gran majoria és informació extreta d'internet i per tant esta clarament esbiaixada.

Justificació de l'ús del model: Decidim utilitzar GPT-2 ja que destaca per la seva capacitat d'entendre el context i realitzar tasques específiques amb pocs exemples, el que fa que sigui una eina potent per processar textos mèdics i classificar informació rellevant en aquest domini amb eficàcia i adaptabilitat. Addicionalment, al tenir models de diferents mides ens permet experimentar més amb els costos i resultats d'aquestes sense haver de canviar significativament el codi.

10.2.2 GPT-Neo

GPT-Neo, o "Generative Pre-Trained Transformer Neo", és un model desenvolupat per EuletherAI, una empresa sense ànim de lucre nascuda d'un servidor de Discord on es parlava de GPT-3 i que ha esdevingut un institut d'investigació sense ànim de lucre líder en la recerca d'intel·ligència artificial a gran escala. Actualment, el seu objectiu principal s'ha convertit en la recerca de la interpretabilitat de l'IA aprofitant la millora substancial de l'accés públic als models

d'IA pre-entrenats a gran escala[12].

Aquest llenguatge és el primer LLM, entrenat amb el conjunt de dades *Pile*, com a intent de la empresa de crear un llenguatge *open source* similar a GPT-3[13]. Es presenta en 3 mides diferents; 125M, 1.3B i 2.7B de paràmetres. Concretament, nosaltres utilitzarem la versió de HuggingFace.

La particularitat de GPT-Neo en contraposició amb GPT-2 és que utilitza atenció local a cada capa amb més de 256 tokens. Això implica que el model es centra en un subconjunt de les dades d'entrada, sovint determinat per un model d'alineaments après. Aquest enfocament és menys costós computacionalment però introdueix operacions no diferenciables[14].

Forts del model:

- Comprensió del context. De la mateixa manera que GPT-2, GPT-Neo captura bé la informació contextual.
- Aprenentatge de transferència. Altre cop aquest factor és important ja que ens permet aprofitar el llenguatge per a tasques de classificació específiques tot i que no siguin les que formen part del seu entrenament original.
- Compatibilitat amb *Zero-shot learning* i *Few-shot learning*. Possibilitat d'utilitzar una quantitat nul·la o gairebé nul·la d'exemples per a realitzar prediccions.
- Entrades variables. Apte per a longituds de text variables.

Debilitats del model:

- Recursos computacionals. Encara més limitat que a GPT-2 ja que la majoria dels models son d'una mida superior.
- Sobre parametrització. L'augment dels paràmetres pot ser no necessari i pot afectar negativament al recursos computacionals que tenim.
- Risc de sobre ajustament. De la mateixa manera que GPT-2, un model gran amb un conjunt de dades petit (en relació) podria provocar un sobre ajustament a les prediccions.

Justificació de l'ús del model: Utilitzem GPT-Neo per tal de poder comparar amb un model similar els resultats. A la seva vegada, aquest model també disposa de diferents mides que ens permeten experimentar amb les prediccions i veure gràficament si el fet de tenir la particularitat de l'atenció local afecta o no als resultats d'aquestes. Un altre motiu de pes, és el fet de que amb el model GPT-Neo podem reutilitzar gairebé tot el codi del anterior model, únicament hem d'importar i utilitzar les llibreries i funcions necessàries.

10.3 *Overview* de les estratègies d'aprenentatge

Per a cadascun dels models utilitzarem dos estratègies d'aprenentatge diferents, *Zero-shot learning* i *Few-shot learning*. Aquestes estratègies son un bon enfoc que permet als LLM realitzar tasques per a les que no han estat específicament entrenats o han estat entrenats però amb un conjunt de dades d'entrenament molt limitat.

10.3.1 *Zero-shot learning* o ZSL

Zero-shot learning és una estratègia en què el model és capaç de realitzar tasques sense haver estat exposat prèviament a exemples d'entrenament específics d'aquesta tasca. Ens és especialment útil ja que en una tasca nova, sense haver vist exemples específics d'entrenament per a aquesta tasca, el LLM pot generar respostes o classificar informació relacionada amb aquesta tasca. Únicament haurem de facilitar-li un *prompt*, una petita explicació de que consisteix la seva tasca, i amb les eines que té l´LLM intentara completar aquesta feina.

En conclusió, utilitzar una estratègia d'aprenentatge combinada amb un LLM com GPT-2 ofereix flexibilitat i adaptabilitat per abordar noves tasques amb poca o cap supervisió específica per a aquestes tasques. Això pot ser beneficiós en escenaris com el nostre ja que el cost de tot l'etiquetatge de dades pot ser molt costos i els recursos computacionals poden no estar al nostre abast.

10.3.2 *Few-shot learning* o FSL

Few-shot learning és una estratègia en què els *Large Language Models* poden realitzar tasques específiques amb un nombre molt petit d'exemples d'entrenament, aportant aquests com a *few shots* d'informació per a la tasca. Els models utilitzen aquesta informació limitada per adaptar-se ràpidament a tasques específiques proporcionant adaptabilitat i flexibilitat a la vegada que probablement una major precisió que en el cas del *Zero-shot learning*[15].

En resum, *Few-shot learning* és similar en quant a el que ens ofereix respecte a *Zero-shot learning* tot i que els pocs exemples que li proporcionarem poden ser crítics depenent de la tasca a realitzar i pot ser una bona pràctica el provar amb ambdós mètodes d'aprenentatge per experimentar en quant a com d'important és la informació que li proporcionem als models abans de la execució d'aquests.

10.4 Implementació del model

A continuació parlarem de com hem implementat els models i farem un breu resum de les decisions preses durant el procés.

10.4.1 Selecció del model

El primer pas de la implementació és seleccionar quin model utilitzarem. Com el nostre objectiu final és el d'experimentar amb diferents mides i models en triarem varis i anirem anotant els resultats per a una futura comparació.

GPT-2

1. Importarem *GPT2LMHeadModel* de la llibreria *transformers* de Hugging Face. Això ens permet utilitzar el model de GPT-2 per a la classificació de les substàncies farmacològiques.

2. Importem *GPT2Tokenizer* que és el *tokenizer* especific per al model GPT-2. Això ens permetrà tokenitzar les nostres oracions o el que és el mateix, convertir text en seqüències d'identificadors numèrics (o "tokens") que el model GPT-2 pot comprendre. Aquesta conversió és necessària perquè els models de llenguatge com GPT-2 operen amb dades numèriques en lloc de text.

3. Utilitzem la funció *from_pretrained()* tant del model com del *tokenizer* i li especifiquem quina versió de la funció utilitzarem; 'gpt2-small', 'gpt2-medium', 'gpt2-large' o 'gpt2-xl'.

4. Es defineixen els tokens de final de seqüència ('eos_token_id'), que indica quan acaba una seqüència i on comença una altra, i de final de màscara ('pad_token_id'), que s'utilitza per afegir "padding"assegurant que totes les seqüències tinguin la mateixa longitud, per al model.

GPT-Neo

1. Importarem *GPTNeoLMHeadModel* de la llibreria *transformers* de Hugging Face. Això ens permet utilitzar el model de GPT-Neo per a la classificació de les substàncies farmacològiques.

2. Importem *GPT2Tokenizer* que tot i que és el *tokenizer* específic de GPT-2 funciona d'igual manera per a GPT-Neo. Això ens permetrà tokenitzar les nostres oracions per a que el model GPT-Neo les pugui comprendre.

3. Utilitzem la funció *from_pretrained()*, exactament igual que amb GPT-2, tant del model com del *tokenizer* i li especifiquem quina versió de la funció utilitzarem; 'gpt-neo-1.3B', 'gpt-neo-2.7B', o 'gpt-neo-125M'.

4. Es defineixen els tokens de final de seqüència ('eos_token_id'), que indica quan acaba una seqüència i on comença una altra, i de final de màscara ('pad_token_id'), que s'utilitza per afegir "padding"assegurant que totes les seqüències tinguin la mateixa longitud, per al model.

10.4.2 Càrrega de les dades

Aquest pas variara lleugerament depenent de quina estratègia d'aprenentatge estem utilitzant ja que les dades necessàries variaran.

ZSL
Carreguem el conjunt de dades de test sencer ja preprocessat el qual només conte una llista de sentencies sense ningun exemple ni resposta. Ho guardem en una llista de cadenes on cada cadena és una de les sentencies per tal d'utilitzar-ho més endavant.

FSL
Carreguem un conjunt de dades de test amb molt pocs exemples de com han de ser les respostes del model, aquestes seran unes poques sentencies les quals tindran marques a cadascuna de les substancies farmacològiques seguint el format indicat en apartats anterior per això, i un altre conjunt de dades de test amb la resta sense respostes. Ho guardem seguint el mateix format que en ZSL.

10.4.3 Creació del *prompt*

El *prompt* és una petita frase o instrucció que dones al model com a entrada per indicar-li quina tasca o comportament concret vols que realitzi. L'ús d'un *prompt* amb FSL o ZSL és una tècnica que ajuda a guiar el model de llenguatge en l'execució de tasques específiques o la generació de contingut particular amb només uns pocs (o cap) exemples d'entrada. Aquesta aproximació és útil quan vols que el model faci alguna cosa concreta sense passar per una formació exhaustiva. La formulació correcta del *prompt* i la complexitat d'aquest juguen un paper molt important a l'hora d'aconseguir que la qualitat de l'aprenentatge sigui la esperada.

El prompt utilitzat en el nostre cas, "For each of the following sentences, mark in XML which are the mentioned drugs and their type(drug, group, brand, drug not usable on humans).", és una petita explicació formulada en mode de ordre de com volem el format de les nostres respostes. Cal destacar que ha d'estar formulat en anglès ja que els models estan pre-entrenats en aquest llenguatge i seria dificultar-lis més la tasca el canviar el llenguatge d'aquest.

Per al cas de ZSL aquest *prompt* serà exactament el que li passarem al nostre model. En canvi, per al cas de FSL, és necessari passar a la primera iteració de les prediccions el conjunt de dades d'exemple que havíem carregat prèviament. Aquest formaran part del *prompt* ja que son una petita explicació exemplificada de com dur a terme la tasca descrita en aquest.

10.4.4 Generació de text per a cada entrada

En aquesta secció del codi recorrerem cada línia del fitxer d'entrada, el conjunt de dades de test, amb l'objectiu de generar text predictiu basant-se en la configuració proporcionada. Per a cada iteració del codi seguirem els següents passos:

1. Es crea un text que combina el *prompt* sencer amb la línia del fitxer d'entrada a predir.
2. S'estableix una mida màxima per a la generació del text, concretament es limita la quantitat de tokens que el model pot generar com a sortida. La mida vindrà en funció de la llargada de la sentencia original i hi afegim 50 ja que creiem que és suficient com per marcar correctament les substàncies farmacològiques pertinents.
3. Es converteix el text complet en identificadors de tokens amb el tokenizer.
4. S'utilitza el model per a generar text predictiu. Marquem amb dos asteriscs on estarà la part de la predicció que fa referència a la resposta esperada per tal de que sigui més fàcil després recuperar el resultat.
5. Es mesura i guarda el temps que es triga en fer la predicció. Això ens serà útil per a la comparació de models i magnituds d'aquests.
6. Es descodifica el text predit utilitzant el tokenizer per eliminar els tokens especials.
7. El text generat es guarda per a ús posterior.

Al finalitzar la generació de text per a totes les entrades guardem el resultat, tant de les prediccions com dels temps, en un fitxer nou de text per a un possible ús posterior.

10.5 Implementació de l'avaluació dels models

A continuació, explicarem pas a pas com hem dut a terme l'avaluació dels diferents models. La implementació d'aquest apartat és comú per a tots els models i per a les diferents estratègies d'aprenentatge.

10.5.1 Càrrega dels conjunts de dades: predit i esperat

En aquest primer apartat farem referencia a la funció *load_predicted_and_expected_data()* que té com a objectiu el carregar i processar les dades des de dos arxius de text diferents, un que conté les dades esperades, *full_test_dataset.txt*, i l'altre que conte les dades predites, *generated_text_MODEL* on MODEL fa referencia al model que estem utilitzant en cada moment.

En resum la funció segueix els següents passos:

1. Estableix les rutes dels arxius d'entrada (*expected_path* i *predicted_path*).
2. Llegeix les dades esperades des de l'arxiu *full_test_dataset.txt* i les emmagatzema en una llista anomenada *expected_dataset*. Aquestes dades estan estructurades com a entrades, on cada entrada és una seqüència de línies.
3. Llegeix les dades predites des de l'arxiu *generated_text_MODEL.txt* i les emmagatzema en una llista anomenada *predicted_dataset*. Només guarda les línies que es troben entre dues línies que contenen asteriscs (**). Aquesta part pot ser utilitzada per excloure certes seccions del text que no ens interessen com poden ser prediccions innecessàries.
4. Elimina les parts no desitjades de les dades predites. És probable que el model intenti predir dades de més ja que és molt complicat explicar-li en aquest quan ha de parar de predir a més de decidir correctament un *max_length* que ens permeti obtenir respostes a la vegada que no es passa de paraules. Per aquesta raó, si processa més d'una línia seguida que comença per la expressió "Response: ", ignora la línia ja que interpreta que es tracta d'una predicció extra no necessària.
5. Crida la funció *pre_process_predicted_and_expected_data()* amb les dades predites i les dades esperades com a paràmetres.

10.5.2 Preprocessat de les dades

A continuació realitzarem un petit preprocessat de les dades que ja hem carregat, és a dir dels conjunts de dades predites i de les esperades, per tal de que sigui més senzilla la posterior comparació entre elles. Per a fer-ho seguirem els següents passos:

1. Aplana la llista *predicted_dataset*, que conté les dades predites, convertint-la d'una llista de llistes a una llista única amb l'ús de la funció *chain.from_iterable*.

2. D'aquesta llista aplanada, crea una nova llista anomenada *responses_predicted_dataset* que conté només les línies amb índexs imparells. Això ho fem per tal de seleccionar només les línies amb les respostes, ja que fins ara contenia tant les sentencies inicials com aquestes segones.

3. Realitza el mateix procés amb la llista *expected_dataset*, que conté les dades esperades.

4. D'aquesta llista aplanada, crea una nova llista anomenada *responses_expected_dataset* que conté només les línies amb índexs imparells per la mateixa raó que ja hem exposat.

5. Finalment, crida la funció *càlcule_results()* amb les llistes de respostes esperades i predites com a paràmetres per realitzar càlculs o processaments addicionals.

10.5.3 Càlcul dels resultats

Per al càlcul dels resultats utilitzarem 3 avaluadors diferents que calculàrem per a cadascuna de les diferents etiquetes (drug, drug_n, brand i group):

- **Precisió.** Consisteix en la proporció d'instàncies classificades com a positives que són realment positives respecte al total de les instàncies classificades com a positives(veritablement positives i falsos positius). Es calcula com a (veritablement positives) / (veritablement positives + falsos positius) o el que és el mateix COR / ACT on COR = veritablement positives i ACT = (veritablement positives + falsos positius).

- ***Recall.*** És la proporció de les instàncies positives que has estat identificades correctament pel model respecte al total de les instàncies positives (veritablement positives i falsos negatius). Es calcula com (veritablement positives) / (veritables positives + falsos negatius) o el que és el mateix COR / POS on COR és el que ja hem dit i POS = (veritables positives + falsos negatius).

- ***F1-Score.*** És la mitjana harmònica entre la precisió i el *recall*, és útil ja que proporciona una mètrica que té en compte tant els falsos positius com els falsos negatius. Es calcula com (2*P*R) / (P+R) on P = precisió i R = *recall*.

Addicionalment, també calculàrem els falsos positius i els falsos negatius de cada una de les etiquetes. Per fer-ho simplement haurem d'aplicar les següents equacions un cop ja haguem obtingut els avaluadors:

- **Falsos positius.** fp = ACT - COR

- **Falsos negatius.** fn = POS - COR

Per a calcular els avaluadors descrits primerament calcularem COR, respostes correctes (el que és el mateix que veritables positius o tp), POS, possibles respostes i ACT, respostes actuals. Per a fer-ho seguirem els següents passos:

1. Inicialitzem els comptadors de COR, POS i ACT per a cadascuna de les etiquetes.

2. Iterem sobre les línies de resposta esperades i predites utilitzant la funció zip que ens permet fer-ho simultàniament.

3. Utilitzem expressions regulars per tal d'extreure el contingut dins de les etiquetes <group>, <drug>, <drug_n>o <brand>.

4. Comparem les coincidències. per a això anirem compten les coincidències que trobem entre les respostes esperades i predites i anirem actualitzant els comptadors de POS i ACT segons la longitud de les coincidències trobades a les respostes esperades i predites.

5. A partir del nombre d'etiquetes que haguem generat en la llista d'esperades extrèiem quantes possibles etiquetes de cada tipus n'hi ha. Això fa referència al POS de cada etiqueta prèviament descrita.

6. De la mateixa manera que a l'apartat anterior, a partir del nombre d'etiquetes que haguem generat en la llista de predites extrèiem quantes actuals etiquetes de cada tipus n'hi ha. Això fa referència al ACT de cada etiqueta prèviament descrita.

7. Comparem un a un els resultats per tal de trobat coincidències exactes, en les que han de coincidir tant la droga en qüestió com la seva etiqueta, per a cada una que trobem incrementem un comptador de *correct* i eliminem la coincidència trobada a les respostes esperades per tal de no comptabilitzar-ho més d'una vegada.

8. Actualitzem el comptador d'encerts COR de cada etiqueta sumant el valor del comptador *correct* que li correspon.

9. Calculem els valors finals de P, R, F1, fp, fn, tp fent ús de les formules descrites anteriorment i els comptadors que hem obtingut.

10.6 Resultats amb ZSL i FSL

A continuació, farem una comparativa dels diferents models provats en la qual tindrem en compte la complexitat dels models, temps, els diferents avaluadors, i estratègies d'aprenentatge utilitzades.

10.6.1 Complexitat dels models

La complexitat dels models fa referencia a la quantitat de paràmetres que té cadascun d'ells. Inicialment intentarem treballar amb GPT2-Small, GPT2-Medium, GPT2-Large, GPT2-XL, GPTNeo-Small, GPTNeo-Medium i GPTNeo-Large i anirem veient la viabilitat de cadascun d'ells. A la Figura 15 podem veure la comparativa entre els diferents models.

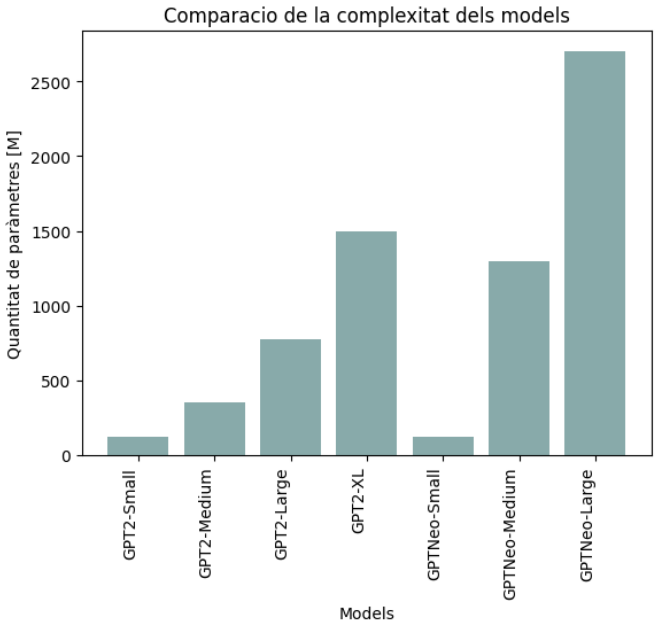

Figura 15: Gràfic amb la comparativa de la complexitat dels models

10.6.2 Temps dels models

A continuació mostrarem els resultats en quant a temps del nostre estudi sobre el temps d'execució per a la predicció de les substancies farmacològiques fent ús dels diferents models i estratègies d'aprenentatge. Cal tenir en compte que els temps els quals hem utilitzat per a la comparativa fan referencia només a les parts de la execució en la que s'estava computant la predicció i que per tant els temps del processat de les dades, l'avaluació d'aquestes i altres no han estat comptabilitzats per tal de no esbiaixar els resultats. A més, s´ha fet ús en tot moment d'un mateix conjunt de dades amb 202 possibles prediccions i sempre

s'ha fet ús del mateix *hardware* i *software* per a la mateixa raó (mantenir la reproductibilitat dels experiments).

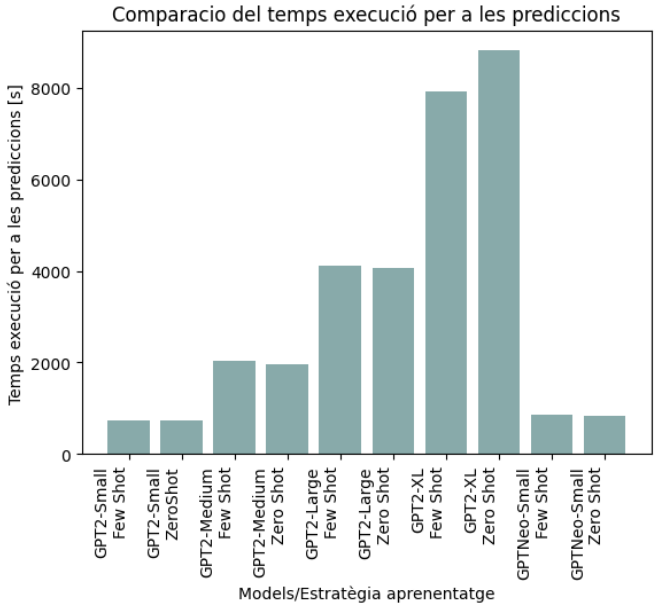

Figura 16: Gràfic amb la comparativa dels temps d'execució

Com podem observar a la Figura 16 els temps es mantenen molt similars entre els mateixos models i diferents estratègies d'aprenentatge. Tot i així, es pot apreciar que en els models més petits pot semblar que *Zero Shot Learning* necessita de menys temps que *Few Shot Learning* però que quan la quantitat de paràmetres augmenta això canvia i passa ser al reves. Tot i així, no tenim suficients exemples ni proves per afirmar aquestes hipòtesis i ho deixarem per a una futura comprovació.

En aquest gràfic ja hem deixat enrere alguns models com GPTNeo-Medium i GPTNeo-Large. Això ha estat degut a que el *software* que estem utilitzant, Google Collab, no té suficients recursos computacionals per executar aquests models amb les nostres especificacions i per tant hem decidit apartar-los i prosseguir amb la investigació amb la resta de models.

10.6.3 COR i ACT dels models

Tot i provar amb diferents especificacions per als models i les estratègies d'aprenentatge els resultats han estat decebedors. De les possibles prediccions que teníem el rati d'encerts ha estat nul. Al fer un estudi de que passava vam poder observar que les respostes que obteníem no eren res més que una copia exacte

de la frase que li proporcionàvem inicialment, fent-nos així creure que el model no havia aconseguit entendre el que li demanàvem.

	tp	**fp**	**fn**	**#pred**	**#exp**	**P**	**R**	**F1**
brand	0	0	276	0	276	0.00%	0.00%	0.00%
drug	0	0	2141	0	2141	0.00%	0.00%	0.00%
drug_n	0	0	60	0	60	0.00%	0.00%	0.00%
group	0	0	688	0	688	0.00%	0.00%	0.00%
average	0	0	3167	0	3167	0.00%	0.00%	0.00%

Taula 7: Taula resum resultats *GPT2* amb ZSL

	tp	**fp**	**fn**	**#pred**	**#exp**	**P**	**R**	**F1**
brand	0	0	276	0	276	0.00%	0.00%	0.00%
drug	0	0	2141	0	2141	0.00%	0.00%	0.00%
drug_n	0	0	60	0	60	0.00%	0.00%	0.00%
group	0	0	688	0	688	0.00%	0.00%	0.00%
average	0	0	3167	0	3167	0.00%	0.00%	0.00%

Taula 8: Taula resum resultats *GPT2* amb FSL

A les taules 7 i 8 podem apreciar que els resultats obtinguts son els mateixos. Això, sembla ser que és degut al fet de que que en cap cas el model ha aprés correctament i per tant simplement té el mateix comportament en ambdós.

10.6.4 Conclusió dels models

La conclusió que podem extreure d'aquesta primera part de la experimentació és que els models per si sols no han estat capaços amb cap de les dues estratègies d'aprenentatge d'aconseguir els resultats esperats. Els motius d'aquest fracas poden ser diversos; des de a la mida dels models que no els hi permet aprendre suficientment bé, a el fet de que els models no han estat específicament ideats per a aquest tipus de tasques, a que el *prompt* no ha estat suficientment clar, o al nombre d'exemples el qual li hem aportat.

Al veure aquests resultats i decidint que efectivament no eren satisfactoris ens decantem per fer un *fine-tuning* dels models ja que creiem que pot ser una estratègia per millorar el rendiment del model en tasques específiques. Alguns motius pels quals l'afinament podria ser beneficiós:

1. Adaptació a la tasca específica: Els models de llenguatge pre-entrenats com GPT-2 i GPT-Neo es formen amb grans quantitats de dades generals.

No obstant això, a vegades, les tasques específiques poden requerir un *fine-tuning* per adaptar-se a la terminologia, l'estil i les particularitats de la tasca en qüestió.

2. Dades limitades: ZSL i FSL sovint depenen de dades limitades per aprendre la tasca específica. L'afinament permet que el model s'ajusti millor a les dades disponibles per a la tasca, capturant patrons específics i millorant el rendiment.

3. Ajustament dels paràmetres: Durant l'afinament, s'ajusten els paràmetres del model per adaptar-se millor a la tasca específica. Això pot incloure ajustaments a les capes finals del model perquè es especialitzi en la tasca en qüestió.

4. Millora de la generalització: L'afinament pot ajudar el model a generalitzar millor per a la tasca específica, especialment quan ZSL i FSL tenen dificultats per generalitzar adequadament a causa de la manca d'exemples d'entrenament.

5. Ajustament d'hiperparàmetres: Durant l'afinament, també es poden ajustar els hiperparàmetres del model per trobar la combinació òptima per a la tasca específica. Això pot incloure la taxa d'aprenentatge, la mida del lot i altres paràmetres relacionats.

11 *Fine-tuning*

11.1 *Overview*

El *fine-tuning* es refereix a un procés en el qual un model d'aprenentatge automàtic preentrenat es reajusta per a realitzar una tasca específica o treballar amb un conjunt de dades concret. Aquest procés implica prendre un model prèviament entrenat en una tasca més general i continuar l'entrenament amb dades addicionals específiques de la tasca que es vol abordar.

En el context de models de llenguatge, com GPT-2 o GPT-Neo, el *fine-tuning* implica prendre un model preentrenat amb grans quantitats de text i ajustar-lo perquè es comporti millor en una tasca o conjunt de dades específics. Aquest procés és útil quan es disposa de dades específiques de la tasca d'interès i es busca millorar el rendiment del model en aquesta tasca en concret.

Durant el *fine-tuning*, els paràmetres del model s'ajusten perquè el model pugui adaptar-se millor a la tasca específica o al domini d'interès. Això pot incloure ajustos a les capes finals del model o fins i tot a capes intermèdies segons la complexitat de la tasca.

En el nostre cas, la experimentació fent ús de l'afinament, consistira en primerament escollir uns hiperparàmetres lògics que ens permetin obtenir uns resultats de la funció de pèrdua satisfactoris i a partir d'aquí farem un *fine-tuning* per a cadascuna de les mides dels models. En aquest cas, i ja que ens dona més joc per les diferents mides amb les quals tenim la capacitat d'experimentar, ens centrarem en utilitzar el model GPT-2 (amb les mides *small* i *medium* ja que son les úniques que suporta Google Collab amb els seus recursos computacionals). També experimentarem amb la mida del conjunt d'entrenament que utilitzem a l'afinament per tal d'investigar fins a quin punt val la pena ampliar la informació que li donem al model a canvi de necessitar uns recursos computacionals i un temps majors.

11.2 Hiperparàmetres

Primerament, farem una petita introducció dels diferents hiperparàmetres que utilitzarem i com afecten aquests a la experimentació.

11.2.1 Nombre d'èpoques

El nombre d'èpoques es refereix al nombre total de vegades que el model itera sobre el conjunt sencer d'entrenament durant el procés d'entrenament. És un hiperaparàmetre molt important que ens permet determinar quant de temps estarà el model entrenant-se e influencia sobre l'habilitat del model per a capturar patrons i generalitzar la informació que encara no ha vist. Configurar el numero apropiat d'èpoques és molt important per tal de prevenir el sobreajustament i el infraajustament.

11.2.2 *Batch size* o mida del lot

La mida del lot fa referència al nombre d'exemples d'entrenament que s'utilitzen en una iteració per actualitzar els pesos d'un model durant el procés d'entrenament. En altres paraules, durant cada pas d'entrenament, el model no processa tot el conjunt de dades d'entrenament, sinó només un subconjunt d'exemples definit per la grandària del lot. La grandària del lot és un hiperparàmetre important que afecta la velocitat d'entrenament i la quantitat de memòria necessària.

És important configurar-ho correctament ja que afectara directament al rendiment computacional, a més gran la grandària del lot més es pot accelerar l'entrenament ja que el model realitzara càlculs per a diversos exemples alhora, aprofitant la capacitat paral·lela de les unitats de processament. Per altre banda, tampoc pot ser massa gran ja que la grandària del lot afecta també a la quantitat de memòria necessària per emmagatzemar els gradients i actualitzar els pesos del model. Grandàries de lot més grans poden requerir més memòria, la qual cosa pot ser limitant en maquinari amb recursos limitats.

11.2.3 Estratègia d'avaluació

Es tracta de la manera com es duu a terme l'avaluació del model durant l'entrenament en el context de l'aprenentatge automàtic. La manera més comú és fer-ho per a *steps*, el que implica que l'avaluació del model es realitzarà segons un nombre específic de passos d'entrenament prèviament indicats. Altres estratègies típiques poden ser per èpoques, que realitza l'avaluació després de cada època completa d'entrenament o directament no fer us de cap avaluació.

11.2.4 Mida del bloc

La mida del bloc és la grandària de bloc utilitzada per dividir les seqüències de text en fragments més petits durant el preprocessament de les dades abans de l'entrenament del model. La mida del bloc pot ser important sobre tot per

motius com la segmentació del text, això és especialment útil quan el text és massa llarg per processar-se d'una sola vegada a causa de limitacions de memòria o recursos computacionals, per a el control de la longitud de la seqüencia, fixar la grandària del bloc controla la longitud màxima de les seqüències d'entrada que el model processarà, per a la memòria del model, blocs més grans poden requerir més memòria, i en el procés de la informació contextual per al model, blocs més petits poden limitar la quantitat de context que el model considera a cada pas.

11.2.5 Hiperparàmetres escollits

En aquesta secció explicarem els hiperparàmetres que hem escollit per a fer el *fine-tuning* i perquè els hem triat.

Hiperparametre	**Valor**
Nombre d'èpoques	5
Mida del lot	2
Estratègia d'avaluació	"steps"
Nombre de passos	500
Mida del bloc	128

Taula 9: Taula resum relació valor per hiperparametre

Com podem observar a la Taula 9 el nombre d'èpoques s'ha establert a 5. Ens ha semblat un numero adequat ja que ens sembla un bon punt de partida i no hem vist una millora considerable al augmentar-les. Per un altra banda, la mida del lot esta configura en 2 degut a les restriccions de memòria que tenim ja que altrament, al intentar augmentar aquest valor, es disparava el temps de computació necessari. Aquesta configuració és la mateixa tant per a l'entrenament com per a l'avaluació. Per a la estratègia d'avaluació hem triat la estratègia per passos i l'hem configurada en 500 passos. El triar aquest nombre ens permet una avaluació més freqüent que ens facilita la monitorització del progres durant la estratègia d'entrenament. Finalment, la mida del bloc esta configura en 128. Un altre cop aquesta configuració ve determinada per els recursos limitats i és una elecció pràctica degut a la longitud mitjana de les seqüencies d'entrenament que ha estat prèviament calculada.

11.3 Procés d'entrenament

11.3.1 Implementació

Per a implementar el nostre *fine-tuning* seguirem els següents passos:

1. Definim el model i el tokenitzador del paquet transformers des dels quals realitzarem l'afinament.
2. Configurem el tokenitzador perquè detecti el final de seqüència fent ús del 'eos_token'. Ens serà útil per tal de gestionar adequadament la delimitació de les seqüències del text.
3. Carreguem els conjunts de dades, tant el d'entrenament com el de validació.
4. Creem el *data collator*. Aquest objecte s'encarrega de reunir i organitzar les dades per al modelatge de llenguatge.
5. Definim els diferent arguments per a l'entrenament. Aquí definim des de les rutes dels directoris de sortida i d'entrada com els hiperparàmetres que utilitzarem.
6. Definim l'objecte *trainer*. Aquest és el que es responsabilitzara del procés d'entrenament. S´hi proporcionen el model, els arguments d'entrenament, el *data collator* i els conjunts de dades d'entrenament i avaluació.
7. Guardem el model ja ajustat. El guardem tant localment per a l'ús pròxim com externament per tal de fer una copia de seguretat que puguem reutilitzar futurament.

11.3.2 *Loss function*

Overview

Per als nostres models fem ús de dos tipus de *Loss function*, la d'entrenament i la de validació, que s'utilitzen amb objectius diferents.

La funció de pèrdua del conjunt d'entrenament s'utilitza per guiar el procés d'ajustament del model. L'objectiu és minimitzar aquesta funció de pèrdua durant l'entrenament perquè el model aprengui patrons específics en les dades d'entrenament i s'adapti a la tasca específica per a la qual es realitza el *fine-tuning*. Aquesta funció és calcula comparant les prediccions generades pel model amb les seqüències reals de paraules al conjunt d'entrenament. El minimitzar la funció de pèrdua durant l'entrenament implica que el model millora la seva capacitat per generar seqüències de text coherents i rellevants per a la tasca específica.

Per un altre banda, la funció de pèrdua del conjunt de validació s'utilitza per avaluar el rendiment general del model en dades no vistes durant l'entrenament. D'aquesta manera el conjunt de validació actua com a conjunt d'avaluació independent i proporciona una mesura objectiva de la capacitat del model per generalitzar a noves dades. Com en aquest cas només proporcionem un conjunt de validació que només conté frases i no respostes, el model està generant text condicional a aquestes frases durant la fase d'avaluació. En aquest cas, l'objectiu no és comparar les prediccions amb respostes correctes específiques, ja que no es proporcionen respostes correctes al conjunt d'avaluació. En canvi, al tractar-se de generació de text, la funció de pèrdua mesura la discrepància entre la distribució de probabilitat predita pel model per a la següent paraula de la seqüència i la distribució de probabilitat real de la paraula següent al conjunt d'entrenament. La biblioteca transformers maneja automàticament aquest procés d'avaluació autoregressiva durant l'ajustament. El valor de la *validation loss* en aquest context es refereix a una mètrica específica (com l'entropia creuada) que mesura la discrepància entre les prediccions del model i el text generat durant la validació. S'espera que una baixa pèrdua al conjunt de validació indiqui que el model generalitza bé les dades no vistes i que ha après patrons útils sense sobreajustar-se al conjunt d'entrenament.

GPT2-Small Full data

El primer *fine-tuning* que realitzarem serà amb tot el conjunt de validació que se'ns ha proporcionat i començàrem provant amb el model petit de GPT-2.

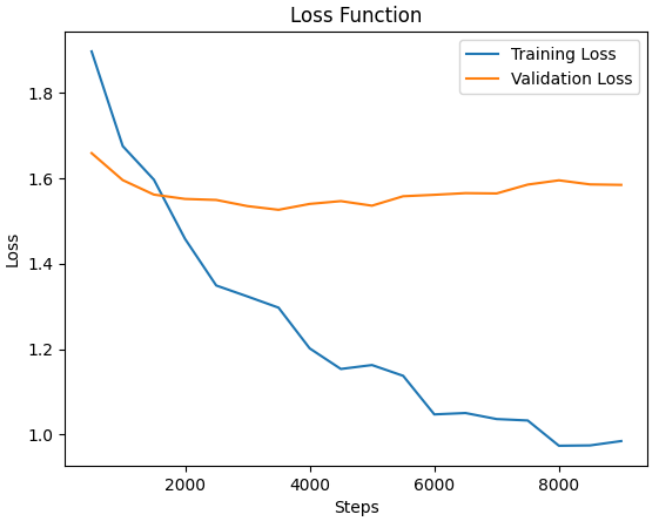

Figura 17: Gràfic de les funcions de pèrdua.

Com podem observar a la Figura 17 la funció de pèrdua de validació es

manté bastant constant mentre anem augmentant els passos mentre que per un altra banda la funció de pèrdua d'entrenament clarament disminueix. En els primers passos (0-1000) sembla que hi ha una gran disminució de la funció de pèrdua. Aquesta pot venir donada pel fet de que el model aprèn i ajusta els pesos per tal de minimitzar les distàncies entre els valors predits i els actuals valors. Això indica que el model està gradualment millorant el seu rendiment. Dels passos 1000 al 2000 veiem clarament que aquesta funció segueix disminuint però a una menor velocitat. El model està afinant encara més les seves prediccions i reduint l'error general. Del pas 3000 cap endavant podem veure algun repunt en la funció de pèrdua del model però no és res més que reajustaments per tal de seguir millorant ja que no és fins al pas 8000 que veiem un possible petit empitjorament i que per tant finalitza l'ajustament.

En resum, podem dir que el model millora ràpidament inicialment i tot i que el ritme de millora disminueix poc a poc aquesta segueix constant.

GPT2-Medium Full data

El segon cas seguira sent amb tot el conjunt de validació proporcionat i fent ús de la mida mitjana del model GPT2.

Figura 18: Gràfic de les funcions de pèrdua.

A la Figura 18 podem veure que la funció d'error d'entrenament i la de validació divergeixen bastant entre elles. Això segurament és degut al fet de que per a la funció d'entrenament la comparació si que és realitzava amb un conjunt de dades correcte mentre que la funció de validació utilitzava un conjunt de mètriques que feien una aproximació al resultat, facilitant així molt més que

l'error augmentes. Igualment, també és probable que si hi hagi un petit risc de sobre-ajustament de les dades però creiem que caldrà veure els resultats finals per estar-ne del tots segurs. Per al que fa a la funció de pèrdua del gràfic veiem que el cas és molt similar al que hem parlat a l'apartat anterior. Inicialment disminueix amb més velocitat i poc a poc la millora va ralentitzant-se i tinguen alguns repunts.

GPT2-Small Half data

En aquest cas utilitzarem exactament la meitat de sentencies per a l'entrenament respecte a les utilitzades en els exemples anteriors. A més, tornarem a fer ús de GPT2 amb la seva mida més petita.

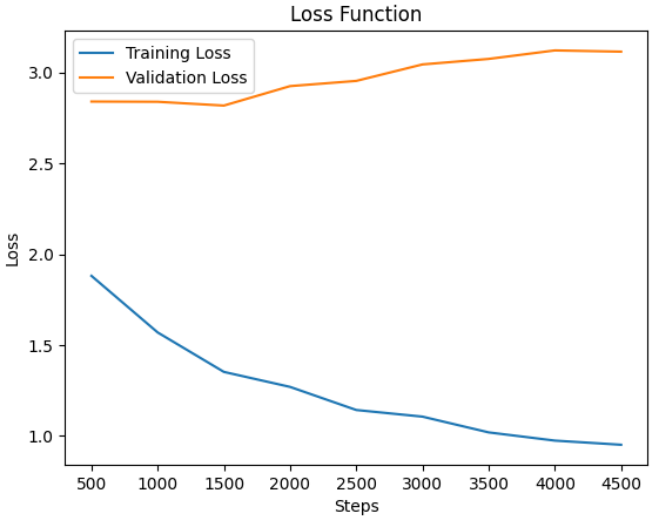

Figura 19: Gràfic de les funcions de pèrdua.

A la Figura 19 podem veure un dibuix que ens recorda al que ja havíem comentat amb *GPT2-Medium Full data* on les funcions d'error divergien bastant. Tot i així, podem observar que en aquest cas la funció de validació és molt més estable i sembla que manté més el seu valor. Per l'altre banda, la funció d'error d'entrenament és molt més similar a l'anterior i la única diferencia notable que apreciem és el fet de que no hi ha repunts tant marcats com a l'anterior exemple. El més probable és que això sigui degut a que en aquest cas estem tractant amb molt menys passos, la meitat, degut a que la mida del conjunt de dades també és la meitat, i per tant tenim menys mostres que en l'anterior. Els errors generalment també son superiors en aquest cas.

GPT2-Medium Half data

Per a aquest cas seguirem utilitzant el mateix conjunt d'entrenament, la meitat del inicial proporcionat, i *GPT2-Medium*.

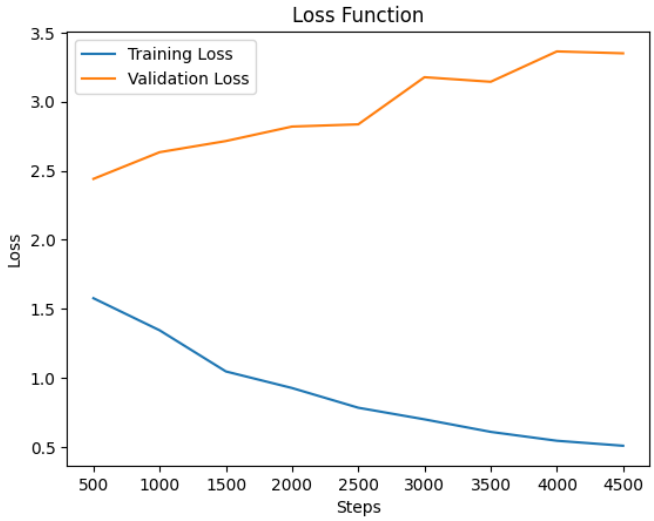

Figura 20: Gràfic de les funcions de pèrdua.

A la Figura 20 veiem que encara que el dibuix de la gràfica és molt similar, en quant a forma a la mida més petita del model, hi ha certes diferències notables. Per començar, en el cas de la funció de pèrdua de validació veiem que aquesta és molt menys estable i que tot i que comença més petita acaba obtenint un valors més grans que l'anterior. Això podria ser un indicador de que el sobre-ajustament en aquest cas és major que a l'anterior. Tot i així, també veiem que l'error d'entrenament és significativament més petit que l'anterior i manté la mateixa curvatura de disminució que ja teníem, acabant així, amb un error més petit, al augmentar els passos, que en el cas anterior.

GPT2-Small Quarter data

A continuació, farem ús d'una quarta part del conjunt d'entrenament i *GPT2-Small* per a l'estudi.

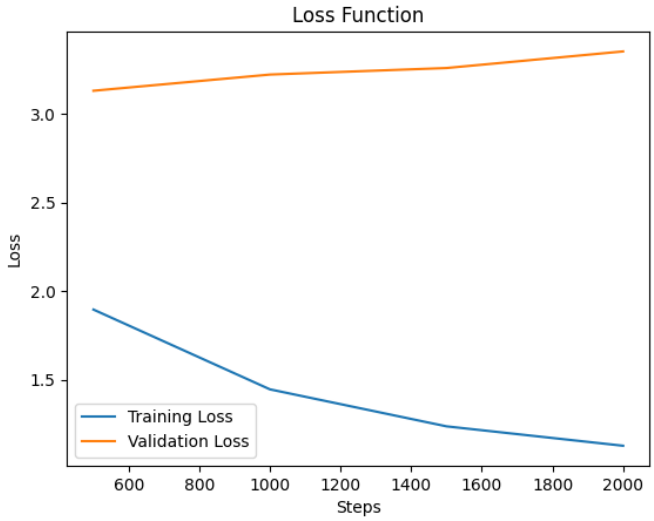

Figura 21: Gràfic de les funcions de pèrdua.

Com podem observar a la Figura 21 la funció de l'error de validació és encara més elevada que amb *GPT2-Small Half data*, tal i com es podria esperar a l'estar utilitzant menys informació per aprendre. De la mateixa manera, a l'haver-hi menys dades, el gràfic també és menys representatiu i s'hi aprecien menys repunts que en els anteriors. En quan a la funció de l'error d'entrenament també és lleugerament superior però és manté molt similar tant en magnitud com en forma. Caldrà veure els resultats de les mètriques del model per veure en quant difereix amb el mateix model amb una base de dades més gran per al seu entrenament.

GPT2-Medium Quarter data

Finalment, acabarem amb la versió mitjana de GPT2 i fent ús d'una quarta part del conjunt d'entrenament.

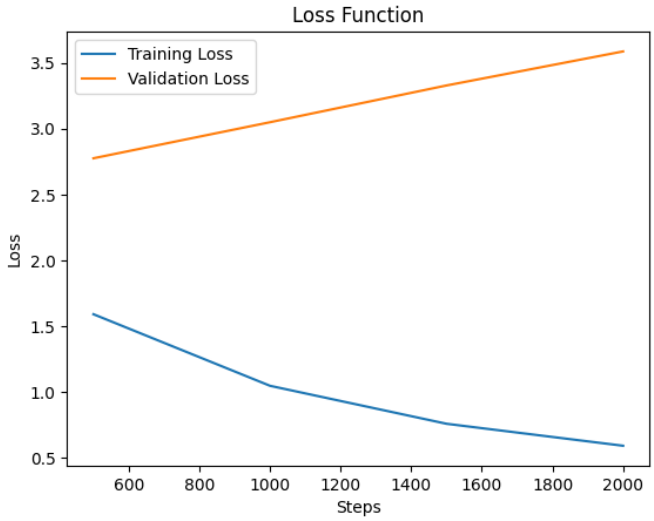

Figura 22: Gràfic de les funcions de pèrdua.

A la Figura 22 veiem que curiosament l'error de la funció de pèrdua de validació sembla correspondre a una funció lineal positiva perfecte. A més, podem veure que la funció d'error és menor tant per a l'error de validació com per a l'error d'entrenament si ho comparem amb el model més petit amb les mateixes dades. A part d'això, la corba de la funció d'error d'entrenament sembla ser idèntica a l'anterior.

11.4 Resultats

En aquesta secció mostrarem els resultats per a tots els models amb els diferents *fine-tunings* realitzats. A la seva vegada, també farem referència als temps de computació necessaris per a cada un dels diferents models i farem una comparativa entre ells per tal de veure si l'augment dels recursos implica un augment del rendiment i precisió del model per tal de verificar la hipòtesis inicial. Finalment, farem una comparativa gràficament entre els resultats dels diferents models.

11.4.1 *GPT2-Small Full data*

Per a *GPT2-Small Full data* obtenim els següents resultats:

- **Temps total per a les prediccions.** Un total de 10103.05 segons o el que és el mateix 2 hores i 48 minuts.
- **Temps mitja per a les prediccions.** 7.06 segons.
- **Precisió del model.** Precisió d'un 72.56%.
- ***Recall* del model** *Recall* d'un 75.45%.
- ***F1-Score* del model.** *F1-Score* d'un 73.98%.

11.4.2 *GPT2-Medium Full data*

Per a *GPT2-Medium Full data* obtenim els següents resultats:

- **Temps total per a les prediccions.** Un total de 29330.85 segons o el que és el mateix 8 i 9 minuts.
- **Temps mitja per a les prediccions.** 20.496 segons.
- **Precisió del model.** Precisió d'un 81.69%.
- ***Recall* del model** *Recall* d'un 82.85%.
- ***F1-Score* del model.** *F1-Score* d'un 82.27%.

11.4.3 *GPT2-Small Half data*

Per a *GPT2-Small Half data* obtenim els següents resultats:

- **Temps total per a les prediccions.** Un total de 10034.69 segons o el que és el mateix 2 hores i 47 minuts.
- **Temps mitja per a les prediccions.** 7.01 segons.
- **Precisió del model.** Precisió d'un 66.79%.
- ***Recall* del model** *Recall* d'un 70.29%.

- ***F1-Score* del model.** *F1-Score* d'un 68.49%.

A la Taula 10 podem veure els seus resultats desglossats depenent de la etiqueta en qüestió.

	tp	**fp**	**fn**	**#pred**	**#exp**	**P**	**R**	**F1**
brand	130	60	146	190	276	68.42%	47.10%	55.79%
drug	1664	714	484	2371	2141	69.89%	77.39%	73.45%
drug_n	0	5	62	5	62	0.00%	0.00%	0.00%
group	432	328	249	767	688	56.32%	62.79%	59.38%
average	2226	1107	941	3333	3167	66.79%	70.29%	68.49%

Taula 10: Taula resum resultats *GPT2-Small Half data*

Com podem observar, els millors resultats els generen les etiquetes amb més possibles valors. Això és degut a que de la mateixa manera que ara tenim més exemples, en el conjunt d'entrenament també hi hauria un percentatge major de valors amb aquestes etiquetes permeten al model aprendre més sobre aquestes. En canvi, la etiqueta drug_n, que compta amb 0 encerts té una quantitat molt petita de mostres proporcionalment a les altres i per tant difícilment el model sabrà identificar-les.

11.4.4 *GPT2-Medium Half data*

Per a *GPT2-Medium Half data* obtenim els següents resultats:

- **Temps total per a les prediccions.** Un total de 29188.755 segons o el que és el mateix 8 hores i 6 minuts.
- **Temps mitja per a les prediccions.** 20.39 segons.
- **Precisió del model.** Precisió d'un 75.36%.
- ***Recall* del model** *Recall* d'un 78.69%.
- ***F1-Score* del model.** *F1-Score* d'un 76.98%.

A la Taula 11 podem veure els seus resultats desglossats depenent de la etiqueta en qüestió.

	tp	fp	fn	#pred	#exp	P	R	F1
brand	191	58	85	249	276	76.70%	69.20%	72.76%
drug	1824	478	317	2303	2142	79.24%	85.19%	82.11%
drug_n	2	33	60	35	60	5.71%	3.33%	4.21%
group	475	246	213	720	689	65.88%	69.04%	67.42%
average	2492	815	675	3307	3167	75.36%	78.69%	76.98%

Taula 11: Taula resum resultats *GPT2-Medium Half data*

De la mateixa manera que el comentat a l'apartat anterior, els resultats es mantenen en la mateixa tendència però, com caldria esperar, veiem una millora en tots els seus avaluadors.

11.4.5 *GPT2-Small Quarter data*

Per a *GPT2-Small Quarter data* obtenim els següents resultats:

- **Temps total per a les prediccions.** Un total de 9880.825 segons o el que és el mateix 2 hores i 44 minuts.
- **Temps mitja per a les prediccions.** 6.9025 segons.
- **Precisió del model.** Precisió d'un 52.56%.
- ***Recall* del model** *Recall* d'un 57.15%.
- ***F1-Score* del model.** *F1-Score* d'un 54.76%.

A la Taula 12 podem veure els seus resultats desglossats depenent de la etiqueta en qüestió.

	tp	fp	fn	#pred	#exp	P	R	F1
brand	69	52	207	121	276	57.02%	25.00%	34.76%
drug	1471	1234	670	2705	2142	54.38%	68.71%	60.71%
drug_n	0	1	60	1	60	0.00%	0.00%	0.00%
group	270	347	418	617	689	43.76%	39.24%	41.38%
average	1810	1634	1355	3444	3167	52.56%	57.15%	54.76%

Taula 12: Taula resum resultats *GPT2-Small Quarter data*

En aquesta taula, tornem a veure com a la etiqueta drug_n no es troba cap coincidència i com els percentatges majors als avaluadors segueixen corresponent als valors amb més intents. Cal destacar que en aquest model el *recall* i el *F1-score* han baixat unes quantitats considerables, sobretot en la etiqueta de brand. Tinguen en compte que la baixada del *F1-score* vindrà donada en gran mesura per la del *recall*, això ens sembla indicar que el nombre de falsos positius ha augmentat significativament.

11.4.6 *GPT2-Medium Quarter data*

Per a *GPT2-Medium Quarter data* obtenim els següents resultats:

- **Temps total per a les prediccions.** Un total de 20716.06 segons o el que és el mateix 5 hores i 45 minuts.
- **Temps mitja per a les prediccions.** 14.62 segons.
- **Precisió del model.** Precisió d'un 70.03%.
- ***Recall* del model** *Recall* d'un 70.48%.
- ***F1-Score* del model.** *F1-Score* d'un 70.25%.

A la Taula 13 podem veure els seus resultats desglossats depenent de la etiqueta en qüestió.

	tp	**fp**	**fn**	**#pred**	**#exp**	**P**	**R**	**F1**
brand	117	35	159	152	276	76.97%	42.39%	54.67%
drug	1720	667	421	2387	2141	72.06%	80.34%	75.97%
drug_n	0	8	60	8	60	0.00%	0.00%	0.00%
group	395	245	293	640	688	61.72%	57.41%	59.49%
average	2232	955	933	3187	3167	70.03%	70.48%	70.25%

Taula 13: Taula resum resultats *GPT2-Medium Quarter data*

En aquest cas, veiem que respecte al mateix model, en la seva versió més petita però amb la mateixa quantitat de dades, hem aconseguit estabilitzar mínimament el *recall* i conseqüentment el *F1-score*. Tot i així,les tendències d'errors i d'encerts segueix siguen similars a les anteriors.

11.4.7 Comparativa dels models

A continuació veurem un seguit de comparacions gràfiques entre els diferents models i les comentarem.

Començàrem amb els gràfics de temps, tant el temps total de predicció, Figura 23, com el temps mitja de predicció, Figura 24, per tal de tenir una visió general de com de complex ha estat el fer ús de cada un dels models.

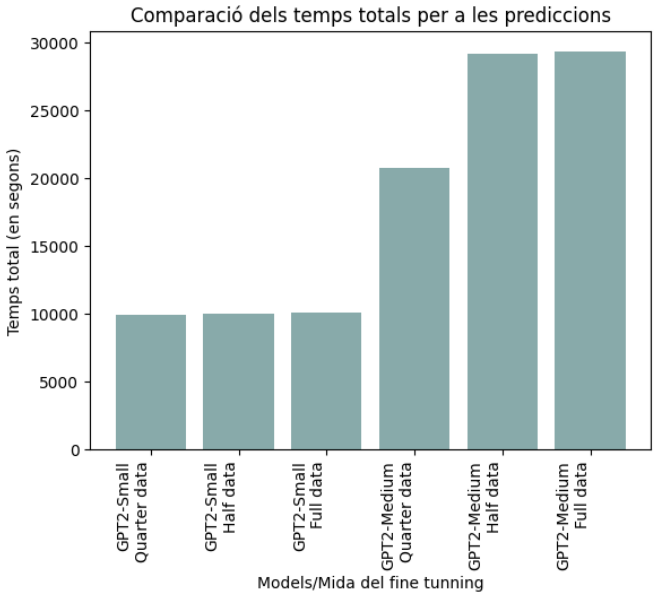

Figura 23: Gràfic de la comparativa dels temps totals per a les prediccions.

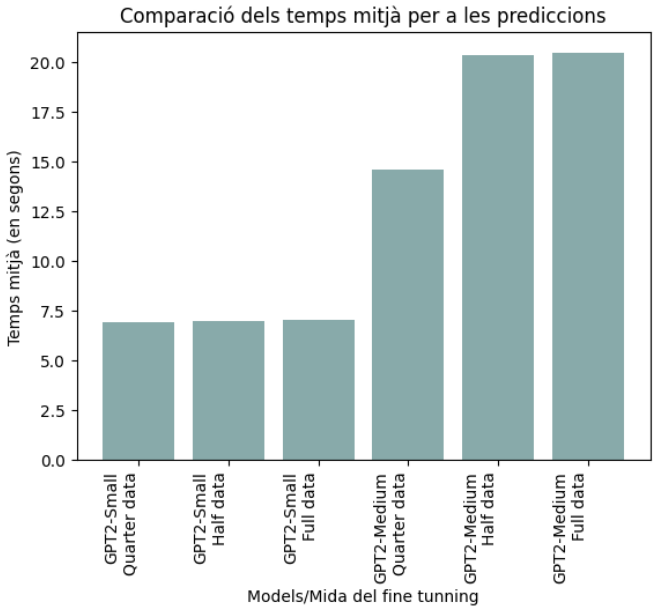

Figura 24: Gràfic de la comparativa de la mitja dels temps per a les prediccions.

Els gràfics de les Figures 23 i 24 han estat prèviament ordenats segons les mides dels diferents models i la quantitat d'informació que els hi proporcionem. Primerament, es mostren els models més petits i amb menys dades, continuen aquest models més petits però augmentant la quantitat de dades, a continuació, tenim els models amb complexitat més gran que de la mateixa forma comencen amb una menor quantitat de dades i després van augmentant. Aquest ordre l'hem fet per poder tenir una millor perspectiva a l'hora de comparar, ja que el que és lògic és que a major informació i complexitat del model major sigui el temps de comput. El que hem de veure a continuació, és valorar si aquest augment en el temps és proporcional al canvi dels resultats posteriorment obtinguts.

Com podem observar la hipòtesis plantejada sobre la relació entre la complexitat i el conjunt de dades utilitzat respecte el temps de comput sembla ser certa. Tot i així, cal tenir en compte que la diferència de la complexitat del model sembla tenir un pes major a l'hora d'augmentar el temps de comput que no pas la mida del conjunt de dades.

El següent gràfic que veurem, a la Figura 25, mostra la precisió, definida com a (veritablement positius) / (veritablement positius + falsos positius), per a cadascun dels casos.

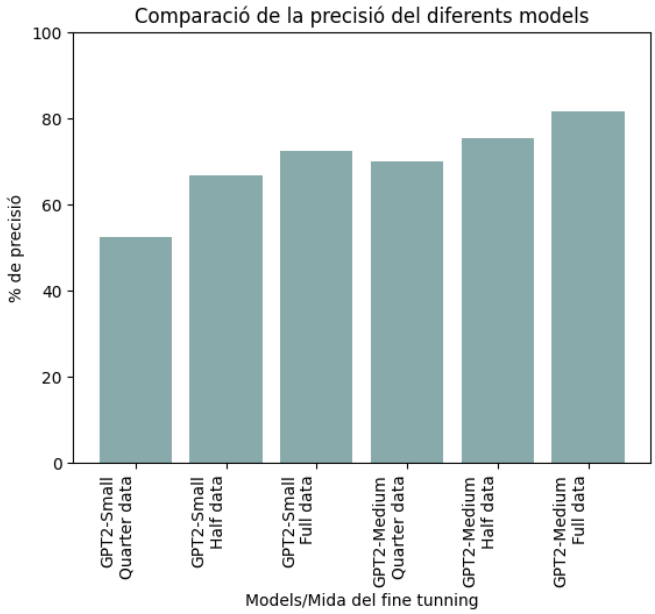

Figura 25: Gràfic de la comparativa de la precisió del models.

Clarament, els millors resultats del gràfic de la Figura 25 els trobem amb *GPT2-Medium* amb el conjunt de dades d'entrenament sencer, mentre que els pitjors resultats, com també caldria esperar, son els de *GPT2-Small* amb una quarta part del conjunt de dades. Cal fixar-s'hi en que els resultats en quant a precisió per a *GPT2-Small* amb tot el conjunt de dades son més favorables que no pas els resultats de *GPT2-Medium* amb un quart del conjunt de dades d'entrenament, el qual li ha més que doblat el temps de predicció. Aquest fet es prou important ja que ens dona peu a plantejar la hipòtesis de que igual si que és més importat fer servir una quantitat de dades major per entrenar el model, quan aquestes estan disponibles, que no pas utilitzar els recursos computacionals per a utilitzar un model més complex. De la mateixa manera veiem aquesta tendència en els altres models de *GPT2-Small* que encara que si que tenen una precisió menor que el model comentat la diferència no és proporcional al temps de més que tarda aquest model més complex.

Ara comentarem el *Recall*, calculat com a (veritablement positius) / (veritables positius + falsos negatius), dels diferents models.

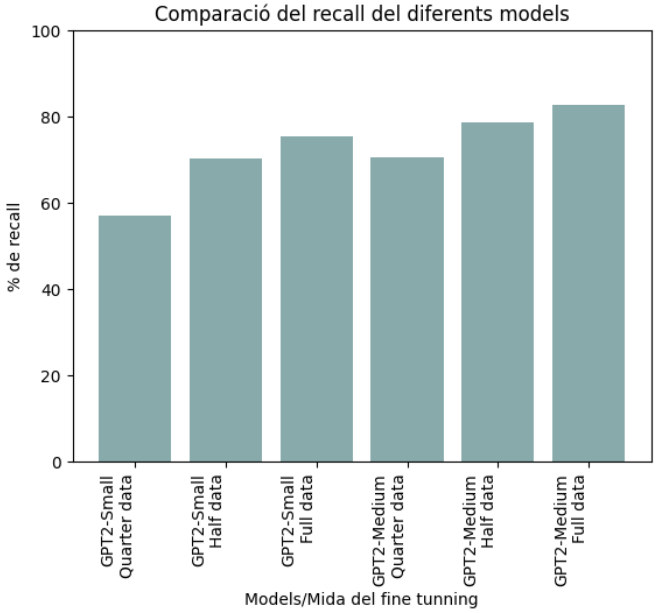

Figura 26: Gràfic de la comparativa del *recall* del models.

En aquest gràfic que tenim a la Figura 26 podem observar com es mantenen les mateixes tendències que veiem a la Figura 25. Per tant, podem deduir que el comportament per al *recall* i per a la precisió dels models és bastant similar. Tot i així, el més destacable és que sembla ser que generalment els percentatges del gràfic del *recall* son lleugerament més elevats el que fa creure que hi ha hagut una quantitat més elevada de falsos positius que de falsos negatius.

Finalment, veurem l'últim apartat que fa referència a l'*F1-Score* calculat com a m (2*precisió**recall*) / (precisió+*recall*). Aquesta mètrica podria ser considerada la de més pes ja que té en compte tots els factors que tenien en compte les anteriors.

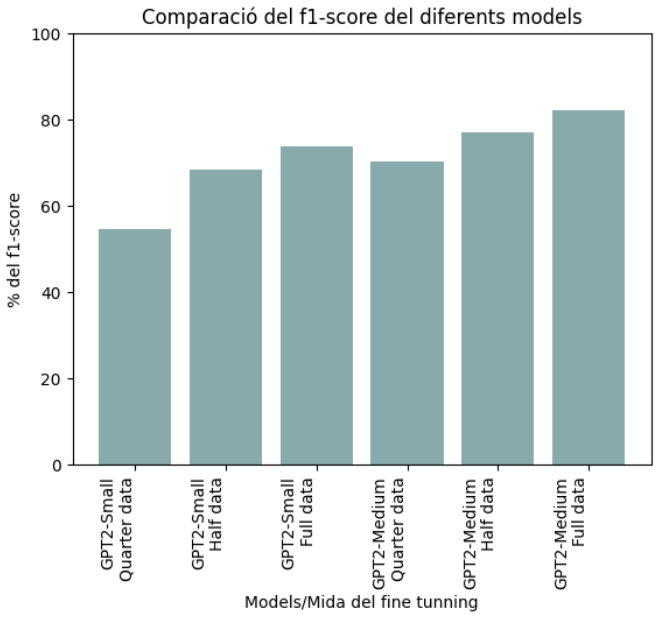

Figura 27: Gràfic de la comparativa del *F1-Score* del models.

A la Figura 27 tornem a observar unes tendències similars a les anteriors. També, es pot observar que amb *GPT2-Medium* amb el conjunt de dades d'entrenament sencer aconseguim un *F1-Score* d'un 82.27%, resultat prou atractiu tinguen en compte els recursos computacionals que estàvem utilitzant i sembla ser que hi ha un bon equilibri entre la capacitat del model per classificar correctament les instàncies positives i evitar falsos negatius i falsos positius.

11.4.8 Conclusió dels models

En resum, els resultats obtinguts en les diferents configuracions de models GPT2 mostren una clara relació entre la complexitat del model, la quantitat de dades utilitzades i el temps de computació necessari. Observem que, en general, els models més grans amb conjunts de dades més extensos tendeixen a proporcionar millors resultats en termes de precisió, *recall* i *F1-Score*. No obstant això, aquest millor rendiment també es tradueix en un augment significatiu en el temps de predicció, tal i com havíem plantejat prèviament.

Per tant, podríem concloure afirmant que la decisió sobre quin model utilitzar i la quantitat de dades necessàries per al *fine-tuning* pot dependre del compromís entre rendiment i recursos computacionals disponibles, destacant la importància de trobar un equilibri òptim entre la complexitat del model i aquesta quantitat de dades per obtenir els millors resultats.

11.5 Conclusions del *fine-tuning*

Després d'experimentar i analitzar els resultats obtinguts amb els models fent ús del *fine-tuning* podem determinar que aquest és clarament beneficiós i que l'especificar la tasca que estàvem realitzant ha estat un paper crucial a l'hora d'aconseguir unes prediccions mínimament fiables. Concretament, hem passat de no obtenir ni tan sols un intent de predicció per a la classificació de substàncies farmacològiques a obtenir fins a un 82.27% de *F1-Score*.

També hem tret conclusions prou significatives en quant a la importància de la complexitat del model i a la grandària del entrenament per al *fine-tuning*. Després de veure tots els resultats, podem concloure amb que és més important la quantitat de data proporcionada que no pas la complexitat del model, tal i com havíem plantejat anteriorment com a hipòtesis. Cal dir, que en certes situacions ens podríem trobar que la informació per a entrenar el model bàsicament no estigues disponible o en tinguéssim una quantitat reduïda. En aquests casos, hem pogut comprovar, que tot i que no son els millors resultats possibles tampoc son del tot nefasts. Per exemple, amb només una quarta part del conjunt de dades però fent ús de *GPT2-Medium* hem obtingut més d'un 70% tant de precisió, com de *recall*, com de *F1-Score*. Per tant, depenent de la necessitat de precisió de la tasca, el reduir el conjunt de dades a la informació de la qual disposem, també pot ser una bona opció.

En conclusió, la mida del conjunt de dades proporcionat per al *fine-tuning* si és important i determinara en gran quantia l'èxit del model. Però per un altre banda, si no es disposa d'una gran quantitat d'informació sempre és millor fer ús de la que tinguem per tal de poder entrenar mínimament el model ja que utilitzant una complexitat major, tot i que necessitem de recursos computacionals majors i més temps, si que és possible aconseguir unes prediccions satisfactòries per a algunes tasques.

12 Conclusions

12.1 *Overview* del desenvolupament del treball

En aquest treball l'objectiu original era fer ús dels models massius del llenguatge per tal d'utilitzar-los en la extracció d'informació utilitzant tècniques com *zero-shot i few-shot learning*. Concretament, volíem utilitzar-los per a la extracció d'informació de textos mèdics i la seva futura classificació, entre drogues aptes pel consum humà, drogues no aptes pel consum humà, marques de medicaments i grups de drogues, de les substàncies farmacològiques prèviament trobades en aquests. Per a fer-ho, hem fet ús del *DDI corpus* el qual consisteix en un conjunt de texts amb informació sobre substàncies farmacològiques i que està creat a partir de dos conjunts de dades independents, *Drug Bank* i *MEDLINE*. En quant als models massius de llenguatge, hem plantejat molts models diferents però finalment hem fet ús de GPT2 i GPTNeo per a la major part de la nostre experimentació. Per un altre banda, les tècniques de *zero-shot i few-shot learning* no han obtingut els resultats esperats amb el que hem decidit fer un posterior *fine-tuning* dels models que ens ha permès obtenir uns resultats satisfactoris i seguir investigant amb aquests. A més, al fer ús del *fine-tuning* també hem dut a terme una experimentació sobre la mida dels conjunt d'entrenament per a aquest on ens hem intentat centrar en anar reduint les dades per veure fins a quin punt la informació que li proporcionem als models era proporcional a els guanys de les mètriques.

En quant al flux de treball, s'han seguit prou fidelment les pautes preestablertes en la part de gestió del treball que és realitzava a l'inici de curs. S'ha realitzat una reunió setmanal amb el tutor per tal d'anar comentant els avenços que teníem i anar fixant noves rutes de treball quan ens trobàvem amb algun problema o decisió a prendre. Cal remarcar que el temps d'experimentació a acabat siguen major del esperat i ha calgut subdividir els conjunts de test en trossos per tal de poder-los processar i posteriorment ajuntar els resultats per a fer el càlcul de les mètriques.

En conseqüència, el desenvolupament del projecte va seguir la següent estructura: la primera part del treball va consistir en la implementació i experimentació amb *zero-shot i few-shot learning*. Al veure que els resultats no tenien la qualitat que esperàvem vam decidir fer una investigació de com aconseguir millorar-los i va ser quan vam decidir dur a terme el *fine-tuning* dels models. La segona part del treball per tant, va consistir en implementar i experimentar amb aquest *fine-tuning* i les possibles combinatòries entre complexitats dels models i quantitat de dades d'aprenentatge. Un cop que vam tenir tota la experimentació vam realitzar els càlculs pertinents per a les mètriques i els gràfics per tal de facilitar la feina de comprensió i extracció de conclusions dels resultats. Finalment, vam acabar de donar els últims retocs a la documentació de la memòria.

12.2 Limitacions, àrees per millorar i futur del treball

La limitació més gran que ens hem trobat amb aquest treball ha estat l'ús de *Google Colab*. Els recursos d'aquesta plataforma son limitats en la versió gratuïta el que ha provocat hores d'inactivitat que podrien haver estat destinades a continuar amb els càlculs. Concretament, els problemes relacionats amb aquesta limitació han estat tres. El primer problema és que les hores de computació continuades per a la plataforma estaven limitades, degut a això hem hagut de segmentar el conjunt de test per a poder realitzar les prediccions ja que altrament és necessitava d'unes 8 hores continuades de les quals no disposàvem. En segon lloc, també vam tenir el problema de que les hores totals diàries, no necessàriament continuades, també estaven capades. Per culpa d'això, per molt que tinguéssim segmentats els conjunt de test, no podíem fer les prediccions amb més de dos d'aquests al dia ja que igualment ens excedíem. El tercer i últim problema de la plataforma era el fet de que els recursos computacionals d'aquesta també son limitats. Això és el causant de que deixéssim d'experimentar amb alguns models, el *fine-tuning* d'aquests excedia els recursos, i que alguns altres no els poguéssim ni provar. En definitiva, l'ús de *Google Colab* ha estat un coll d'ampolla important per a la nostre experimentació que no ens ha permès realitzar una experimentació tant exhaustiva com la que ens hauria agradat.

Per un altre banda, el fet de tenir una manca d'experiència en quant a aquesta temàtica també ha suposat una necessitat d'una major quantitat de temps i ha provocat certs malentesos durant la implementació del projecte. Per exemple, creure que es necessitava fer un *fine-tuning* tant per a *zero-shot learning* com per a *few-shot learning* quan des d'un principi no es necessitava de proporcionar cap exemple com a *prompt*.

Un altre problema amb el que hem hagut de lidiar, ha estat el fet de voler afegir noves estadístiques al final del projecte, concretament el calcular cada avaluador per a cada diferent etiqueta. Tot i que, en gran part, ho hem pogut dur a terme ja que guardàvem els resultats de les prediccions de cada model, sinó hauria suposat una quantitat de temps de la qual no disposàvem, en algun moment del projecte se'ns va perdre la predicció per a *GPT2 full data*. Degut a això, ens ha estat impossible re-calcular tot el model, que hauria suposat unes 16 hores les quals hauríem hagut de repartir en diversos dies, per el que hem hagut d'obviar la taula amb aquests resultats de l'estudi i deixar només els resultats totals.

En quant a àrees per millorar en aquest treball en un futur, si comptéssim amb més temps i recursos ens hauria agradat experimentar amb models massius del llenguatge amb una complexitat molt major i conjunts de dades molt més petits o inexistents, com en el cas del *zero-shot learning*, per veure si d'aquesta forma si era possible aconseguir resultats satisfactoris. També ens hauria agradat estendre a quant més poder la investigació de en quin punt deixa de compensar el augmentar la quantitat d'informació del conjunt d'entrenament

per a la millora de les mètriques. I finalment, també hauria estat interessant el provar amb una quantitat més diversa de models massius de llenguatge que divergissin més de la estructura dels GPT que hem utilitzat.

12.3 Conclusions finals

Aquest projecte tenia com a objectiu principal el aconseguir classificar substàncies farmacològiques d'uns textos mèdics fent ús de models massius de llenguatge i algunes tècniques d'aprenentatge com *few-shot, zero-shot learning* i *fine-tuning*. Els models i les tècniques d'aprenentatge han aconseguit els resultats desitjats i ens han permès treure conclusions a partir dels diferents experiments en els que hem anat fent variar la complexitat dels models i la mida dels conjunts de dades.

Tot i així, encara queda un marge de millora bastant gran que podríem explotar a base d'ampliar la experimentació i afegir nous models amb complexitats millors. A pesar dels inconvenients i problemes que ens hem anat trobant durant el desenvolupament del treball estem satisfets amb els resultats obtinguts i amb la feina feta ja que creiem que el coneixement en aquest àmbit s'ha vist molt ampliat. Tot i que també és cert, que si tornéssim a repetir el treball des de zero, amb aquests nous coneixements, hi hauria moltes decisions que s'haurien pres diferent.

En conclusió, creiem que s'han complert els punts més importants del projecte i s'han dut a terme correctament les alternatives per als problemes trobats i tot i que certament, hi ha millores a fer, l'aprenentatge en aquest ha estat molt satisfactori i hem aconseguit l'objectiu principal de classificar les substàncies farmacològiques a partit de models massius de llenguatge.

13 Bibliografia

[1] Ekin Tiu. Understanding Zero-Shot Learning: Making ML More Human. [en línia] [Consulta: 22 setembre 2023]. Disponible a: https://towardsdatascience.com/understanding-zero-shot-learning-making-ml-more-human-4653ac35ccab

[2] Deval Shah. Few-Shot Learning Guide. [en línia] [Consulta: 22 setembre 2023]. Disponible a: https://towardsdatascience.com/understanding-zero-shot-learning-making-ml-more-human-4653ac35ccab.

[3] Mike Myer. What Is a Large Language Model? [en línia] [Consulta: 23 setembre 2023]. Disponible a: https://www.elastic.co/es/what-is/large-language-models

[4] Tao Tu S. Sara Mahdavi Jason Wei Hyung Won Chung Nathan Scales Ajay Tanwani Heather Cole-Lewis Stephen Pfohl Perry Payne Martin Seneviratne Paul Gamble Chris Kelly Abubakr Babiker Nathanael Sch arli Aakanksha Chowdhery Philip Mansfield Dina Demner-Fushman Blaise Ag uera y Arcas Dale Webster Greg S. Corrado Yossi Matias Katherine Chou Juraj Gottweis Nenad Tomasev Yun Liu Alvin Rajkomar Joelle Barral Christopher Semturs Alan Karthikesalingam Vivek Natarajan Karan Singhal, Shekoofeh Azizi. The future of primary care. [en línia] [Consulta: 23 setembre 2023]. Disponible a: https://www.nature.com/articles/s41586-023-06291-2.

[5] Karthik Shiraly. GPT-J vs. GPT-3. [en línia] [Consulta: 23 setembre 2023]. Disponible a: https://www.width.ai/post/gpt-j-vs-gpt-3

[6] Silvia Solera. Las Mejores Metodologías para un Correcto Desarrollo de Software. [en línia] [Consulta: 24 setembre 2023]. Disponible a: https://www.occamagenciadigital.com/blog/las-mejores-metodologias-para-un-correcto-desarrollo-de-softwarepost-index-1

[7] Conoce el precio de la vivienda en alquiler en tu zona. [en línia] [Consulta: 02 octubre 2023]. Available at: https://www.fotocasa.es/fotocasa-life/alquiler/conoce-el-precio-de-la-vivienda-en-alquiler-en-tu-zona/

[8] ¿Qué es el precio del kWh? [en línia] [Consulta: 02 octubre 2023]. Available at: https://selectra.es/energia/info/que-es/precio-kwh#precio-kwh.

[9] Isabel Segura-Bedmar, Paloma Martínez, César de Pablo-Sanchez, (2011). Üsing Linguistic Kernel for Drug-Drug Interaction Exctraction", Journal of Biomedical Informatic, 44(5), 789-804.

[10] Lluís Padro Cirera.(2023) "The DDI corpus", 1-5.

[11] Hugging Face. "GPT-2 Model Page."[en línia] [Consulta: 1 novembre 2023] Disponible a: https://huggingface.co/gpt2

[12] EleutherAI. "About Us."[en línia] [Consulta: 18 novembre 2023] Disponible a: https://www.eleuther.ai/about

[13] EleutherAI. "GPT-Neo Artifacts."[en línia] [Consulta: 20 novembre 2023] Disponible a: https://www.eleuther.ai/artifacts/gpt-neo

[14] Hugging Face. "GPT-Neo Documentation."[en línia] [Consulta: 20 novembre 2023] Disponible a: https://huggingface.co/docs/transformers/model_doc/gpt_neo

[15] Hugging Face. "Few-Shot Learning with GPT-Neo and the Inference API."[en línia] [Consulta: 30 octubre 2023] Disponible a: https://huggingface.co/blog/few-shot-learning-gpt-neo-and-inference-api