Anàlisi d'algorismes de Record Linkage sobre grafs

Carlos Boned Riera

Resum Una de les branques d'estudi dels Knowledge Graphs són els problemes de record linkage. S'han introduït una gran sèrie d'algorismes que els intenten resoldre de manera efectiva i consistent. En aquest projecte avaluarem diferents tipus d'aquests algorismes, des de mètodes inductius com els que ofereix el Neo4j fins algorismes d'aprenentatge profund, passant per algorismes de machine learning. Tot això amb la finalitat de fer una anàlisi exhaustiva d'aquests enfocaments diferents. Aquestes anàlisis es faran sobre el Dataset del cens de capitals de comarques de Catalunya "BALL"[10]

Paraules clau– Record Linkage, Machine Learning, Deep Learning, Inductius, Estadística, Knowledge Graph, ER, Similarity

1 Introducció

La tasca de descobriment d'enllaços, també coneguda com a *Record Linkage* en aquest camp, és un repte, principalment, a causa de la mala qualitat de les dades i la naturalesa desequilibrada d'aquesta tasca. D'una banda: l'estat de conservació dels documents originals, el procés d'escaneig, el gran nombre de valors similars en noms, edats o adreces són només alguns dels múltiples factors que poden afectar la qualitat de les dades. A més a més, la relació entre els membres de la llar i el cap de la llar pot canviar significativament entre dos censos de registres i, com a resultat, els mètodes d'enllaç de registres no són prou fiables provocant, sovint, que es generin moltes coincidències falses o duplicades [4]. D'altra banda, el descobriment d'enllaços és una tasca desequilibrada, ja que el conjunt d'enllaços candidats creix quadràticament respecte al nombre de nodes individuals, però només uns pocs seran reals.

Els grafs de coneixement (KG) organitzen la semàntica de manera estructurada. Representen una col·lecció de descripcions entrellaçades d'objectes del món real, esdeveniments, situacions o conceptes abstractes, que s'anomenen entitats, en una estructura formal, sent alguns exemples populars freebase, DBpedia, YAGO, Satori, etc. que es componen de milions d'entitats i milers de milions d'enllaços d'entitats. El descobriment d'enllaços, o de manera equivalent, la tasca de trobar enllaços en nodes KG pot ser una tasca difícil en alguns escenaris. Per exemple, en demografia històrica, la reconstrucció de cursos de vida individuals implica vincular dades d'una mateixa persona que apareixen en diferents documents, com el baptisme i el certificat de matrimoni entre d'altres. Aquestes dades es representen naturalment mitjançant KG, on els nodes poden representar persones, esdeveniments, llars, ciutats, etc. i les arestes representen les relacions entre les persones i els esdeveniments ocorreguts a prop en una data determinada.
És per aquesta raó que analitzar els diferents algorismes en aquests escenaris és una tasca molt interessant a realitzar.

1.1 Objectius

L'objectiu principal del projecte és analitzar els principals mètodes per a la resolució de problemes de record linkage i fer-ne una comparació per al dataset BALL. Amb aquesta comparació experimental podrem treure conclusions sobre quin mètode és millor. Per a realitzar aquesta comparativa s'ha complert el propòsit de tenir una representació de les dades adient a l'estructura, tant amb els mètodes més convencionals com amb els mètodes més nous de deep learning. La representació s'esmenta més endavant i ha estat disposada a la BBDD del motor Neo4j. Els diferents objectius són:

- Analitzar i comparar els diferents mètodes de resolució de record linkage, enfocant-se des de mètodes més clàssics de machine learning, fins mètodes de deep learning més complexos.
- Concloure si les característiques de cada experiment són representatives en l'àmbit d'aprenentatge autònom.
- Treure conclusions detallades sobre els diferents mètodes, així com discussió i comparativa de resultats.

• E-mail de contacte: carlos.boned@autonoma.cat
• Treball tutoritzat per: Oriol Ramos Terrades, Ciències de la computació
• Curs 2021/22

2 RELATED WORK

La tasca de descobriment d'enllaços s'ha abordat des de diferents punts de vista i representacions per a KG. La representació més bàsica del KG actualment és el gràfic de relació entitat (ER), els nodes del qual són les entitats i la matriu d'adjacència representen les relacions de les entitats. El mètode TransE [2] i el mètode TransH method [14] són dos enfocaments transductius que busquen incrustacions d'entitats que estan relacionades amb el KG.

La tasca de record linkage ha estat explorada per diverses disciplines, incloses les bases de dades, les estadístiques i la intel·ligència artificial. Cada disciplina ha formulat el problema lleugerament diferent i, en conseqüència, s'han proposat diferents tècniques [13]. A la comunitat de bases de dades, aquesta tasca també es coneix com a deduplicació. La deduplicació té com a objectiu eliminar dades repetides o còpies múltiples i comprimir així la base de dades. Amb aquesta finalitat, s'han suggerit diversos mètodes basats en la distància d'edició de cadenes per a la concordança de registres com un esquema de propòsit general [11] o un enfocament intensiu de coneixement [12].

En estadística, s'ha dut a terme una llarga línia d'investigació sobre l'enllaç de registres probabilístics, basat en gran mesura en el document seminal [6]. Els autors de [6] formulen la concordança d'entitats com un problema de classificació, on l'objectiu bàsic és classificar els parells d'entitats com a concordants o no coincidents. Proposen utilitzar mètodes no supervisats, basats en una representació basada en característiques de parells dissenyats manualment i, fins a cert punt, específics del problema. Tot i que això pot ser un problema important a l'hora d'enllaçar dades de diferents fonts, aquestes propostes han estat, en general, adoptades per investigadors posteriors, sovint amb elaboracions del model estadístic subjacent. La distància Jaro-Winkler ha estat utilitzada per a aquestes propostes d'enllaç de registres [9], [15].

3 METODOLOGIA

Backgroud

Per una banda, tenim els mètodes inductius. En termes generals, un KG és un conjunt de tres bessons compost per dues entitats s i o que representen els objectes (nodes) del KG i les relacions r que representen els enllaços (arestes) entre subjectes (s) i objectes (o). Seguint [3], un KG es defineix com $G = \{(s, r, o) \in \mathcal{D} \subseteq N_e \times N_r \times N_e\}$, on $s, o \in N_e$ són les entitats subjecte i objecte, N_e és el conjunt de totes les entitats existents al KG, r són les relacions KG i N_r el conjunt de tota la relació existent al KG. Els algorismes d'aprenentatge inductiu (ILA) són uns algorismes iteratius d'aprenentatge automàtic que s'utilitza per generar un conjunt generalitzat de classificació, que produeix regles de la forma "IF-THEN", per a un conjunt d'exemples, produint-les a cada iteració i afegint-les al conjunt. Aquest aprenentatge inductiu es realitza per a cada node inici i s'extrapola per a cada un dels seus veïns a través de les relacions entre ells. Sota aquesta idea es genera una sèrie de descriptors de característiques basades en els veïns de primer ordre de cada node. Per als mètodes inductius s'ha fet servir la base de dades Neo4j. El Neo4j segueix un tipus de representació ER, representada en la Fig. 1. Sobre aquesta representació s'han realitzat una sèrie d'experiments per a comparar-los amb els mètodes més clàssics de machine learning. Els dos algorismes utilitzats són el Node2Vec i el GraphSage.

El **node2Vec**[7] és una estratègia de mostreig de *neighborhood* flexible que ens permet interpolar sense problemes entre BFS i DFS. Ho aconseguim desenvolupant un procediment de *random walk* esbiaixat que pot explorar barris en un BFS amb una metodologia DFS.

Formalment, donat un node font u, simulem un random walk de longitud fixa l. Denoteu el node i_{th} a la caminada, començant per $c_0 = u$. Els nodes c_i es generen mitjançant la distribució següent:

$$P(c_i = x | c_{i-1} = v) = \begin{cases} \frac{\pi_{x,v}}{\mathcal{Z}} & si \qquad (x,v) \in E \\ 0 & otherwise \end{cases} \tag{1}$$

On $\pi_{x,v}$ és la probabilitat transitoria no normalitzada entre els nodes *v* i *x*, i $\mathcal{Z}$ és una constant normalitzadora. El Node2Vec ha generat els embeddings per als nodes tipus valor, definits en la Fig. 1. En aquest cas l'algorisme s'utilitza per a tenir una representació de les característiques *[Nom, Cognom, Segon Cognom i Ocupació]*, característiques que utilitzem en els experiments. Tenint una representació de les entitats valor podem definir els embeddings dels nodes individus amb l'algorisme Graph Sage [8]

La intuïció darrere de l'algorisme **GraphSage**[8] és que a cada iteració, o profunditat de cerca, els nodes agreguen informació agregada dels seus veïns locals, i a mesura que aquest procés itera, els nodes guanyen cada vegada més informació del graf. En l'arquitectura d'aquesta agregació, a diferència de l'aprenentatge automàtic sobre *N-D lattic* (per exemple: frases, imatges o volums 3-D), els veïns d'un node no tenen un ordre natural; per tant, les funcions d'agregació de l'algorisme han d'operar sobre un conjunt no ordenat de vectors. Idealment, una funció agregadora seria simètrica (és a dir, invariant a permutacions de les seves entrades) alhora que es pot entrenar i manté una alta capacitat de representació. La propietat de simetria de la funció d'agregació garanteix que el nostre model de xarxa neuronal es pugui entrenar i aplicar al conjunt de característiques de veïnatge de nodes ordenats arbitràriament. La base de l'algorisme radica en l'agregador d'informació que pot ser de diferents maneres, des de un agregador *mean* o fins a un *pooling*. Per als experiments s'ha utilitzat l'agregador pooling. La definició dels dos tipus d'agregadors és:

- **Mean aggregation**: mitjana dels vectors en funció dels elements $h_u^{k-1}, \forall u \in N(v)$
- **Pooling**: Aquest agregador és alhora simètric i entrenable. En aquest enfocament d'agrupació, el vector de cada veí s'alimenta de manera independent a través d'una xarxa neuronal $\mathbf{FC}$. Després d'aquesta transformació, s'aplica una operació d'agrupació màxima per elements i l'agrega a través del conjunt veí.

$$AGGREGAT E_k^{pool} = max(\sigma(W_{pool} h_{u_i}^k + b) \tag{2}$$

$\forall u_i \in \mathcal{N}(v))$

On *max* denota l'operador màxim per element i σ és

una funció d'activació no lineal. En principi, la funció aplicada abans de l'agrupació màxima pot ser un perceptró multicapa arbitràriament profund, però en aquest treball ens centrem en arquitectures simples d'una sola capa.

En cada entrenament s'han definit els nodes individus com entitats, els quals són susceptible a inferència, i els nodes que actuen com a diccionari de tipus valor, que han estat representats amb un vector de característiques amb el Node2vec. Per a les relacions s'ha utilitzat les relacions entre els nodes Individu i els nodes tipus Valor. Les features utilitzades han estat aquesta representació de característiques per al primer experiment. Per al segon experiment s'ha fet servir com a feature *a més a més* el **Cohort** i finalment s'han afegit les característiques d'ocupació i classe històrica.

Funcions de Similitud

Els algorismes descrits anteriorment s'utilitzen per a tenir un vector de característiques que descriuen les entitats esmentades. Per a realitzar les classificacions d'aquestes entitats individus s'han provat diferents mètodes de càlcul de similitud de vectors o càlcul de distàncies entre vectors. Aquests càlculs donen números amb rangs diferents. Per a poder fer la comparativa experimental entre ells s'ha utilitzat el càlcul de l'àrea sota la corva (*auc*), mentre que per a dur a terme les comparatives experimentals amb l'altra branca experimental s'ha continuat utilitzat la mètrica del f-score. Les funcions de similitud del neo4j es divideixen en dues seccions. La primera secció són funcions numèriques que calculen la similitud en funció de la proximitat en la qual es troben els dos vectors de característiques dintre de l'espai geomètric. La segona secció són funcions categòriques que tracten els vectors com a conjunts i calculen la semblança en funció de la intersecció entre els dos conjunts.

Per a comparatives una mica més extenses s'han provat els dos tipus de mètodes i s'han afegit dues mètriques de distàncies manualment:

- **Jensen-Shannon**: La distància Jensen-Shannon calcula la distància entre dues distribucions de probabilitat. Utilitza la fórmula de divergència de Kullback Leibler (l'entropia relativa) per trobar la distància. En aquest cas considerem els embeddings de dos individus com una distribució.
- **Minkowski**: La distància de Minkowski com a generalització tant de la distància euclidiana com de la distància de Manhattan
- **Cosine**: El cosinus distance és una mètrica que s'utilitza per mesurar la semblança dels valors dels diferents registres dels nodes de la base de dades, independentment de la seva mida. Matemàticament, mesura el cosinus de l'angle entre dos vectors, que representen al node, projectats en un espai vectorial. La similitud del cosinus és avantatjosa perquè fins i tot si els dos nodes similars estan molt separats per la distància euclidiana, és probable que encara estiguin orientats més junts. Com més petit sigui l'angle, més gran serà la similitud del cosinus.
- **Euclidean Distance**: La distància euclidiana entre dos punts del pla o de l'espai tridimensional mesura la longitud d'un segment que connecta els dos punts. És la forma més òbvia de representar la distància entre dos punts.

A més a més s'ha utilitzat l'algorisme de **nodeSimilarity**. L'algoritme compara un conjunt de nodes en funció dels nodes als quals estan connectats. Dos nodes es consideren similars si comparteixen molts dels mateixos veïns. Node Similarity calcula les similituds per parelles basant-se en la mètrica de Jaccard, també coneguda com a puntuació de semblança de Jaccard, o en el coeficient de superposició, també conegut com a coeficient de Szymkiewicz–Simpson.[1] Aquest algorisme utilitza les mètriques de similitud categòriques esmentades, on les seues fórmules són:

Jaccard

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{3}$$

Overlap.

$$O(A,B) = \frac{|A \cap B|}{\min(|A|,|B|)} \tag{4}$$

L'entrada d'aquest algorisme és un gràfic bipartit i connectat que conté dos conjunts de nodes disjunts. Cada relació comença des d'un node del primer conjunt de nodes i acaba en un node del segon conjunt de nodes. L'algorisme **NodeSimilarity** compara cada node que té relacions de sortida entre ells. Per a cada node n, recollim el veïnat de sortida $\mathrm{N}(n)$ d'aquest node, és a dir, tots els nodes m de manera que hi hagi una relació de n a m. Per a cada parell n, m, l'algorisme calcula una similitud per a aquest parell que és igual al resultat de la mètrica de similitud seleccionada per a $\mathrm{N}(n)$ i $\mathrm{N}(m)$.

Mètodes de Machine Learning

Els **mètodes de machine learning** són algorismes programats que reben i analitzen dades d'entrada per predir els valors de sortida dins d'un rang acceptable. Aquests algorismes poden ser supervisats (Logistic Regression, LinearSVC), quan les dades estan etiquetades i per cada cas en l'entrenament es diu si hi ha encert o no, i no-supervisats (**ECM clustering**), quan no hi ha aquesta etiqueta. L'aplicació dels models implica realitzar una comparativa de similitud de registres amb l'api de record linkage. En tots els experiments els mètodes necessiten dur a terme una comparativa dels atributs $n \times n$. Siguin els mètodes supervisats definits anteriorment:

Logistic Regression

El mètode de regressió logística és un mètode estadístic que es fa servir per resoldre problemes de classificació binària, on el resultat només pot ser de naturalesa dicotòmica, és a

[1] Aquest algorisme s'ha utilitzat amb els experiments definits pels experts de demografia històrica

dir, només pot prendre dos valors possibles.

$$f(x) = \frac{1}{1 + e^{-k(x-x_0)}} \tag{5}$$

On 1 és el valor màxim que defineix la curva de la sigmoide.

Linear SVC

Intuïtivament, una SVM és un model que representa els punts de mostra a l'espai, separant les classes a dos espais tan amplis com sigui possible mitjançant un hiperplà de separació definit com el vector entre els dos punts, de les 2 classes, més propers al que es diu vector suport.

$$f(x) = \beta + (\mathcal{W} * k(x, y)) \tag{6}$$

On $k(x, y) = x^t \cdot y + c$ i sent c una constant, x l'input i y el vector de suport per separar les dades.

EMCM

L'algorisme de maximització d'expectativa (EM) és un mètode iteratiu per trobar estimacions de màxima probabilitat o màxim a posteriori (MAP) de paràmetres en models estadístics, on el model depèn de variables latents no observades. La idea del model és aconseguir maximitzar una funció de probabilitat definida per l'equació.

$$P(Model|Data) = \frac{P(Data|Model)P(Model)}{P(Data)} \tag{7}$$

Donat un model estadístic que genera un conjunt X de dades observades, un conjunt de dades latents no observades o valors Z que falten i un vector de paràmetres desconeguts , juntament amb una funció de versemblança $L(\theta; X, Z) = p(X, Z|\theta)$, l'estimació de màxima probabilitat (*MLE*) dels paràmetres desconeguts està determinada per la probabilitat marginal de les dades observades.

$$L(\theta; X) = p(X|\theta) = \sum_Z p(X, Z|\theta) \tag{8}$$

L'algorisme EM busca trobar el *MLE* de la probabilitat marginal aplicant iterativament una pasa de d'expectació (les dades (que falten) s'estimen a partir de les dades observades i les estimacions actuals dels paràmetres del model) definida per l'equació:

$$Q(\theta|\theta(t)) = E_{Z|X,\theta(t)}[logL(\theta; X, Z)] \tag{9}$$

I una passa de maximització on s'assumeix que les dades desconegudes se saben:

$$\theta(t+1) = \underset{\theta}{\operatorname{argmax}}[Q(\theta|\theta(t))] \tag{10}$$

Repetir fins la convergencia.

Cal dir que en la Api de record linkage aquest algorisme està definit segons la metodolia esmentada en combinació amb el model de Fellegi i Sunter [5]

A més a més s'ha afegit una metodologia extra on s'aplica un algorisme d'aprenentatge profund. S'ha definit una MLP molt bàsica de 3 capes. Els MLP (MultiLayer Perceptron) són xarxes tipus feed-forward, amb neurones de tipus perceptró. La funció d'agregació és una suma ponderada, i la funció d'activació, un sigmoide, cosa que permet un aprenentatge per backtrack. La MLP està definida per la funció següent:

$$\hat{y} = \text{MLP}^{\psi}(c) = \text{MLP}^{\psi}(\text{SIMILITY}(x_n, x_m)) \tag{11}$$

$\forall n, m \in X$ and $m \neq n$ i on *SIMILITY* implica el vector de similitud de la comparativa dels dos registres amb les característiques definides per cada experiment.

Metodologia Estadística

Per a la comparativa entre els diferents experiments s'ha definit uns tests d'inferència estadística per tenir certesa de si afegir alguna variable nova implica una millora dels mètodes. Per aquest motiu s'ha realitzat un *Pair test* entre els diferents mètodes de manera escalonada. Per a cada algorisme podem definir si per cada experiment hi ha una evidència estadística del fet que hi hagi millora. En aquest cas els resultats estan definits en la TAULA 3

Per a les comparatives entre els 4 algorismes esmentats en la subsecció anterior s'han generat una sèrie de tests estadístics per a tenir certesa si hi ha diferència significativa entre ells. Entre aquests tests es poden definir el test de l'ANOVA. L'ANOVA és un model estadístic conegut per determinar si hi ha una diferència estadística entre diferents distribucions. La hipòtesi nul·la és que les variables funcionen de manera similar i que les diferències només es deuen a l'atzar. Test de tuckey per a diferenciar quina parella de mètodes són significatius entre ells. Finalment, una aproximació de Bayes per a un *correlated test* on s'aproxima probabilísticament el $p\text{-}valor$ per a sebre amb quina probabilitat una model és millor que l'altre[2].

A més a més s'han utilitzat els mètodes de la G-Mean i de la Youdest's J statistic per a aconseguir el millor threshold per a la seva posterior classificació. La mitjana geomètrica (**G-Mean**)[1] és una mètrica que mesura l'equilibri entre els rendiments de classificació tant a les classes majoritàries com a les minoritàries. Una G-Mean baixa és una indicació d'un mal rendiment en la classificació dels casos positius encara que els casos negatius estiguin correctament classificats com a tal. Aquesta mesura és important per evitar l'ajustament excessiu de la classe negativa i l'ajustament insuficient de la classe positiva. La fórmula ve definida per:

$$G - mean = \sqrt{\textbf{sensitivity} \times \textbf{especificity}} \tag{12}$$

L'índex de Youden és una mesura de la corba ROC (Receiver Operating Characteristic). Mesura l'eficàcia i permet la selecció d'un valor llindar òptim (punt de tall) per al marcador.

$$\text{J} = \textbf{sensitivity} \times \textbf{especificity} - 1 \tag{13}$$

Sobre aquests mètodes s'ha hagut de realitzar una normalització de les dades. Aquesta normalització ha estat definida

[2]Aquest valor es calcula a partir de la diferència de les mostres, en aquest cas una mostra conté 10 valors de *f-score* per a cada sampling

per la fórmula següent:

$$x_i^{'} = \frac{1}{1 + e^{-\frac{x_i - \mu_i}{\sigma_i}}} \tag{14}$$

On la μ i σ són la mitjana i la desviació estàndard de cada embedding

4 EXPERIMENTS

Els experiments realitzats han estat tots realitzats sobre les dades del BALL per a poder fer la comparativa amb tots els diferents mètodes, des de els algorismes més clàssics de machine learning fins als algorismes de deep learning utilitzats sobre la BBDD amb el motor *neo4j gds*. Les particions sobre les dades també han estat les mateixes. A més a més, cada branca experimental ha tingut el seu *ablation study* pel que fa a les característiques. Aquest estudi ens dona la importància que tenen aquestes característiques dintre d'aquest problema, d'altra banda, ens aportarà conclusions sobre la informació demogràfica que es pot extrapolar gràcies a aquestes: com l'evolució social, entre altres.

El conjunt de dades demogràfiques del Baix Llobregat (BALL) conté els registres censals de la població de Sant Feliu de Barcelona recollits en 16 censos diferents entre 1828 i 1940. El conjunt de dades conté al voltant de 60.000 registres d'individus amb 30 atributs que augmenten el nombre de nodes al voltant de 140.000. Els atributs disponibles inclouen el nom complet de la persona, l'any de naixement, l'estat civil, l'ocupació i el parentiu amb el cap de família. Representem aquests registres en un EAR KG tal com es mostra a la Fig. 1. El KG conté entitats que representen les llars i els seus habitants. Una vora que uneix aquests dos nodes representa les persones que viuen en una llar en una data determinada. Els nodes quadrats representen els atributs dels nodes i les relacions. Així, l'atribut Únic que vincula els individus I1 i I3 representa l'estat civil entre aquestes dues persones. De la mateixa manera, les vores directes entre persones que viuen a la mateixa llar representen el parentiu amb el cap de família.

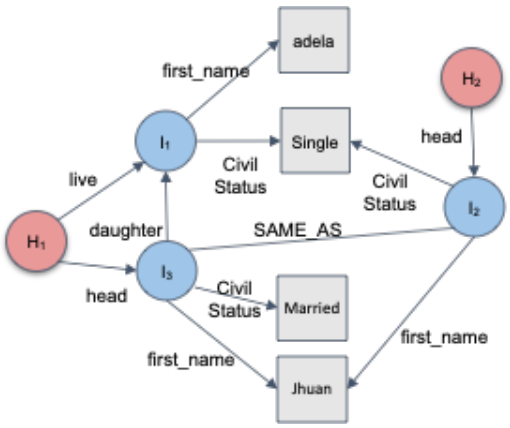

Fig. 1: EAR KG representació per al conjunt de dades BALL. Els nodes arrodonits blaus i vermells representen entitats casa i individu. Els nodes quadrats grisos representen atributs. Les vores *SAME_AS* representen els enllaços que s'han de descobrir a la tasca *Record linkage*.

4.1 Set-up Experimental

La partició experimental ha estat igual en tots els tipus d'algorismes perquè la part comparativa sigui el real possible. Les dades han estat dividides en dos datasets diferents, un dataset A i un dataset B. Per a generar aquests datasets s'han filtrat les dades pel municipi de Sant Feliu del Llobregat i per parelles d'anys consecutius.
Per a l'entrenament s'han agafat aquestes parelles d'anys: (1889, 1906) & (1930, 1936) sent el primer valor de cada tupla els anys que formen part del dataset A i sent el segon valor de cada tupla els anys que formen part del dataset B.
Finalment per la partició de test s'han agafat també persones de Sant Feliu del Llobregat per parelles d'anys diferents, també consecutives, que són: (1910, 1915) & (1924, 1930). Per a reduir el nombre de parelles només s'han agafat aquelles que tinguin el mateix *Segon Cognom*.

Per a cada branca experimental s'han desenvolupat una sèrie de mètodes estadístics. Aquests mètodes estadístics són els encarregats de definir si una característica és prou significativa en l'entrenament del model en qüestió. A més a més s'ha realitzat un test estadístic que compara no només si la diferència dels dos models és significativa o no, sinó que realitza una aproximació de *Bayes* pel mètode de Montecarlo on calcula la probabilitat que un mètode sigui millor que l'altre, o quina és la probabilitat que els dos mètodes siguin igual de bons.
Per a totes les branques d'experiments s'ha realitzat l'estudi de la mateixa manera. S'han definit una sèrie de característiques estratificades. Aquesta estratificació ha estat definida per *l'Institut de Demografia Històrica de Catalunya*. Per al primer experiment de cada mètode només es tindran les característiques [*Nom, Cognom, Segon Cognom*], en el segon experiment s'afegeix l'any de naixement dels individus[3]. En aquest cas les característiques del segon experiment són [*Nom, Cognom, Segon Cognom, Cohort*]. Finalment, s'afegeix el treball i la posició social que aquest treball implica, on el nombre de característiques finals és [*Nom, Cognom, Segon Cognom, Cohort, Ocupació, Classe Històrica*].

S'han definit 3 tipus d'experiments, on a cada experiment se li afegeix una característica més.

Experiment 1

[*Nom, Cognom, Segon Cognom*]: Per aquest experiment només tenim la informació del nom complet de cada individu.

Experiment 2

[*Nom, Cognom, Segon Cognom, Cohort*]: Per aquest experiment s'ha utilitzat la mateixa informació que a l'experiment anterior. No obstant això, ara, a més a més, s'ha afegit la comparativa de l'any de naixement. En aquest cas l'any de naixement ha de ser el mateix per cada individu. Cosa que ens ajuda a establir i comparar cicles vitals.

[3] Aquesta informació està definida com a Cohort

Experiment 3

[*Nom, Cognom, Segon Cognom, Cohort, Job, Historical Class*]: Per aquest últim experiment s'ha afegit l'ocupació, és a dir, el treball d'aquest individu i en quina posició social es troba per fer aquesta feina. Per als valors que no estaven afegits o documentats s'ha establert un valor general de -2 i per a les posicions aristòcrates està definit el -1.

L'estudi ha estat sobre dades no harmonitzades, per a comparar la robustesa dels diferents algorismes.

4.2 Experiments de Machine Learning

Cada experiment del *ablation study* ha estat comparat amb un *Correlated paired t-test* amb un $\alpha = 0.99$. En tots els experiments els mètodes necessiten realitzar una comparativa dels atributs $n \times n$. Per cada experiment s'ha executat l'algorisme 10 cops amb un sampling estratificat del 0.1 i s'han guardat les dades. Finalment, s'ha portat a cap un estudi d'errors per saber si aquest sampling és estadísticament significatiu. En ningun dels casos ha estat així. Per a realitzar les comparatives que necessiten els diferents mètodes de l'Api de record linkage s'han aplicat algorismes de càlcul de similitud entre les variables dels datasets a creuar. Aquests algorismes són els mateixos per als diferents experiments.

- **Experiment 1** [*Nom, Cognom, Segon Cognom*]: Dels algorismes comparatius de strings s'ha fet servir el de jarowinkler[9]. S'ha fet servir aquest mètode per la seva robustesa davant falles de transcripció. Com que les dades no estan harmonitzades, aquestes falles estan present en la base de dades. Sabent això, aquest càlcul no ho penalitza excessivament. Per tant, el percentatge de semblances amb el valor original pot ser molt acurat, depenent del factor de correlació que s'implementa en aquest algorisme. La comparativa amb aquesta similitud s'ha realitzat per totes les variables.

- **Experiment 2** [*Nom, Cognom, Segon Cognom, Cohort*]: Per aquest experiment el càlcul de similitud del nom complet ha estat el mateix que en l'experiment anterior. Per a la comparativa amb el *Cohort* s'ha utilitzat una comparativa d'exactitud, donat que l'any de naixement ha de ser el mateix sempre.

- **Experiment 3** [*Nom, Cognom, Segon Cognom, Cohort, Job, Historical Class*]: Per a l'Experiment 3 s'han provat diferents comparatives. La de millor rendiment ha estat afegint la classe històrica com a variable, és a dir, el valor que hi ha als diferents registres del dataset és el valor que s'afegirà com a comparativa. Finalment pel càlcul de la similitud de l'ocupació he utilitzat el cosinus distance perquè ha estat la que millors resultats ha donat.

El nombre de dades per cada Experiment està expressat en la taula següent:

	train_sampling	true_links	test_sampling	true_links
BALL	7216150	816	201533	8987

TAULA 2: ON CADA TRAIN SAMPLING DEFINEIX UN SAMPLE DELS 10 MENTRES QUE EL TEST ESTÀ COMPLERT

4.2.1 Resultats Estadístics

En aquesta secció estan definits els resultats dels diferents tests estadístics, amb les seves interpretacions pertinents.

	Experiment1	Experiment2	Experiment3
Anova	False	True	True

TAULA 4: TAULA ANOVA SOBRE ELS 3 EXPERIMENTS PER SABER SI HI HA ALGUNA PARELLA DE MODELS ESTADÍSTICAMENT DIFERENTS

Com podem veure en la taula de l'ANOVA, a l'experiment 2 i a l'experiment 3 almenys un model és estadísticament significatiu respecte als altres. Apliquem el test de tuckey per obtenir aquests models i obtenim les taules 5 i 6.

Model1	Model2	diff mitjana	p-adj	inferior	superior	descarta
ECM	Logistic	-0.0113	0.001	-0.0157	-0.0068	True
EMCM	NN	-0.0069	0.001	-0.0114	-0.0025	True
ECM	SVM	0.0006	0.9	-0.0039	0.0051	False
Logistic	NN	0.0043	0.0594	-0.0001	0.0088	False
Logistic	SVM	0.0119	0.001	0.0074	0.0163	True
NN	SVM	0.0075	0.001	0.0031	0.012	True

TAULA 5: EXPERIMENT 2

Model1	Model2	diff mitjana	p-adj	inferior	superior	descarta
ECM	Logistic	-0.0092	0.001	-0.0142	-0.0042	True
EMCM	NN	-0.0111	0.001	-0.0161	-0.0061	True
ECM	SVM	-0.0038	0.1954	-0.0088	0.0051	False
Logistic	NN	-0.0018	0.7344	-0.0068	0.0032	False
Logistic	SVM	0.0055	0.0272	0.0005	0.0105	True
NN	SVM	0.0073	0.002	0.0023	0.0123	True

TAULA 6: EXPERIMENT 3

Aquí tenim el test de Tuckey per als diferents experiments. La columna de "descarta" és la columna que ens dona la informació sobre si s'accepta o no la hipòtesi nul·la. En aquest cas la hipòtesi diu si els dos models són estadísticament diferents. Aquí només estan els models on hi ha hagut de realitzar-se un càlcul per discernir si hi ha una diferència significativa entre els models. Si el càlcul del p-adj està sota un threshold definit per l'api, aquesta comparativa no està afegida a la taula.

No obstant això, el model ens diu que hi ha diferència significativa, però no ens diu amb quina probabilitat un mètode és millor que l'altre. Per a realitzar aquest test s'utilitza una aproximació pel mètode de bayes per al *p-value*. Aquestes probabilitats es defineixen en la taula 1.

4.3 Experiments dels Mètodes Inductius

Per als mètodes inductius hem fet servir la base de dades de Neo4j. Aquesta base de dades té uns algorismes

	Experiment 1			Experiment 2			Experiment 3		
	P(A >B)	P(A=B)	P(A <B)	P(A >B)	P(A=B)	P(A<B)	P(A>B)	P(A=B)	P(A <B)
P(SVM, Logistic)	-	-	-	0.858	0.141	1.51e-7	0.040	0.959	4.35e-5
P(EMCM, Logistic)	-	-	-	0.855	0.144	7.45e-9	0.374	0.625	7.28e-6
P(EMCM, NN)	-	-	-	0.18	0.81	2e-4	0.671	0.328	4.21e-6
P(SVM, NN)	-	-	-	0.215	0.784	1.1e-4	0.221	0.778	2.97e-4

TAULA 1: BAYES CORRELATED TEST ON ESTÀ DEFINIDA LA PROBABILITAT QUE UN MODEL SIGUI MILLOR QUE UN ALTRE. ELS MÈTODES QUE NO SURTEN ÉS PERQUÈ EL SEU VALOR ESTADÍSTIC NO SUPERAVA UN LLINDAR DEFINIT PEL MATEIX MÈTODE I ON L'EXPERIMENT 1 NO HA ESTAT NECESSARI DONAT QUE LA HIPÒTESI NUL·LA DE L'ANOVA NO HA ESTAT DESCARTADA

	Experiment1-Experiment2		Experiment1-Experiment3		Experiment2-Experiment3	
	pValue	Diferència Significativa	pValue	Diferència Significativa	pValue	Diferència Significativa
Logistic Regression	∼0	True	2e-7	True	0.0265	True
SVM	1.37e-09	True	7.38e-08	True	0.008	True
ECMC	8.62e-10	True	9.41e-09	True	>0.05	**False**
NN	3.19e-05	True	7.69e-05	True	>0.05	**False**

TAULA 3: EN AQUESTA TAULA ESTAN DEFINITS ELS RESULTATS DE CADA COMPARATIVA ESTADÍSTICA AMB EL MÈTODE DE PAIRED TEST. SIGUIN LES COMPARATIVES PER TENIR CERTESA SI HI HA MILLORA O NO AFEGINT LES VARIABLES DEFINIDES EN LA SECCIÓ 4.2

incrustats en el mateix motor de la base de dades. Amb aquests mètodes inductius s'han tret representacions de característiques dels nodes respecte al seu subgraf associat. Els principals mètodes utlitzats per a generar aquests embeddings han estat el node2Vec i el Graph Sage. Per a calcular i classificar els possibles nodes han estat utilitzats mètodes de càlcul de similitud entre vectors.
La representació de les dades segueix una idea diferent del tipus de representació que s'ha utilitzat en la branca experimental anterior. En aquella no hi havia representació més enllà dels registres del csv, mentres que dintre del neo4j s'utilitza una representació ER, representada en Fig. 1. Sobre aquesta representació s'han realitzat una sèrie d'experiments per a comparar-los amb els mètodes més clàssics ja definits. Tot i que les característiques emprades són les mateixes, la metodologia és una mica diferent pel mateix comportament intrínsec del motor experimental del Neo4j.
Tot i que la comparativa dels diferents mètodes ha estat realitzada amb els experiments definits pels experts, com que la representació ER afegeix la relació dels diferents registres donada la seva estructura de graf, s'han afegit dos experiments:

Experiment 4

Aquest afegeix la informació de les famílies. En aquest cas afegir les relacions familiars entre els diferents individus afegeix molta inforamció i robustesa sobre els embeddings, creats pel mateix algorisme *GraphSage*

Experiment 5

Finalment, aquest experiment implica afegir tot el graf amb totes les seves relacions. La informació que queda implica l'evolució de la casa, és a dir, si s'ha anat mouen un individu al llarg de la seva vida per diversos habitatges. Aquesta informació ajuda a explorar més el graf i veure els moviments demogràfics dels individus, així com ajudar a refer el cicle vital d'un individu.

Per com està representada la base de dades els números del dataset d'entrenament i de test difereixen tot i que les particions encara són les mateixes. El nombre de dades per a cada experiment és el següent:

	Entities	Relations	Candidate_Pairs	true_links
Experiment1	91753	452098	201533	8987
Experiment2	91753	452098	201533	8987
Experiment3	107794	559130	201533	8987
Experiment4	107877	903056	201533	8987
Experiment5	128964	1294532	201533	8987

TAULA 7: ON LES ENTITATS SÓN ELS NODES CARREGATS EN MEMÒRIA I RELATIONS LES RELACIONS D'AQUESTS PER CADA TIPUS D'EXPERIMENT. L'EXPERIMENT 5 REPRESENTA TOT EL GRAF

4.4 Resultats

Els resultats dels diferents experiments han estat definits sota la mètrica del F-score donat que estem en un problema de classificació.
Per als mètodes de machine learning els resultats han estat directament afegits a la taula comparativa final. El resultat ha estat obtingut realitzant una mitjana dels diferents samplings. Això ha estat possible arrel que els resultats dels tests estadístics sobre els samplings han estat negatius. Això vol dir que no hi ha diferència estadísticament significativa entre els diferents samplings. Els resultats es poden observar en la TAULA 8.
Per als mètodes inductius tenim una taula diferent per a comparar les diferents mètriques de similitud. En la taula podem diferenciar dos tipus de resultats: la *roc auc* i el *f-score*. En aquest cas l'àrea sota la corba (AUC) és la mesura de la capacitat d'un classificador per distingir entre classes i s'utilitza com a resum de la corba ROC. Com més gran sigui l'AUC, millor serà el rendiment del model a l'hora de distingir entre les classes positives i negatives. Els resultats

		ECM	SVM	NN	Logistic	NodeSim	GraphSage+Sim
F-score	Experiment 1	0.949	0.950	0.950	0.950	0.738	0.467
	Experiment 2	0.977	0.978	0.970	0.966	0.750	0.350
	Experiment 3	0.977	0.974	0.966	0.968	0.689	0.401

TAULA 8: LA TAULA MOSTRA ELS MILLORS RESULTATS DE CADA MÈTODE, ON EL NODESIMILARITY I EL GRAPHSAGE SÓN ELS ALGORISMES DE LA BRANCA EXPERIMENTAL DEL NEO4J MENTRE QUE ELS ALTRES SÓN ELS MÈTODES CLÀSSICS DE MACHINES LEARNING. ELS RESULTATS DEL GRAPHSAGE QUE SURTEN A LA TAULA SÓN ELS MILLORS OBTINGUTS AMB LES DIFERENTS MÈTRIQUES DE SIMILITUD

es poden observar en la TAULA 9

Per als mètodes de **Machine Learning** podem afirmar diverses coses. La primera de totes és que afegir característiques sí que millora el rendiment dels diferents algorismes com podem veure en la TAULA 3. Els dos únics algorismes que no es veuen afectats per afegir una característica han estat en el pas de l'experiment 2 al 3. Podem concloure que afegir l'ocupació històrica no ha estat estadísticament significatiu en el mètode no supervisat i en la MLP. A més a més podem veure que a partir de l'experiment 2 hi ha algorismes amb un percentatge més elevat d'encert que altres en una comparativa directa, com ha estat definida en la TAULA 1. Finalment, podem afirmar que pel tipus de dades, realitzar un sampling estratificat no provoca una variació en els resultats estadísticament significativa, cosa que ha provocat una reducció de dimensionalitat per a realitzar els entrenaments. Amb tot això podem veure els resultats dels mètodes de Machine Learning en la taula comparativa amb els mètodes inductius TAULA 8
Respecte als mètodes **Inductius** podem observar en la TAULA 9 que els resultats no han estat del tot satisfactoris. Estadísticament, només podem afirmar que hi ha una diferència significativa entre l'algorisme NodeSimilarity i GraphSage. Cal esmentar que el motor del neo4j ha estat problemàtic per a realitzar els entrenaments. Com que aquests algorismes es troben en fase Beta i Alpha no han estat optimitzats per a la GPU, cosa que ha fet que entrenaments relativament ràpids tardessin un temps molt elevat. Per una altra banda, com que s'utilitza la versió community no es podia optimitzar la memòria, provocant que no es poguessin ni augmentar el nombre de capes ni el nombre de random walks. Aquest fet provoca que el *finetunnig* hagi estat ineficient. Fent una petita anàlisi veiem que la distància de *Shannon-Jensen* no és millor en cap moment ni en cap mena de mètrica, mentre que les altres tenen el seu despuntar. No obstant això, per a fer unes primeres comparatives els resultats han estat satisfactoris en el cas del NodeSimilarity, que per la definició de l'algorisme han estat uns nombres coherents.

5 COMPARATIVES

En la TAULA 8 podem veure els diferents resultats del *f-score* per als diferents experiments amb cada un dels mètodes. Com ja ha estat comentat els mètodes de machine learning han estat millors que els mètodes inductius del motor del neo4j, tenint en compte les problemàtiques esmentades.
Per al ***Experiment 1*** podem veure com tots els mètodes de machine learning són clarament millors que els mètodes inductius. No obstant això, no es pot destacar cap d'ells donat que el seu rendiment ha estat el mateix. En la taula comparativa de Bayes 1 no ha fet falta realitzar el test estadístic perquè no s'havia pogut determinar estadísticament que els diferents mètodes es comportessin diferent. En els resultats podem veure que ha estat així i, per tant, podem veure que hi ha una coherència amb els tests estadístics.
Per al ***Experiment 2*** podem veure com l'algorisme unsupervisat i el SVM són lleugerament millors que tots els altres mètodes. Si veiem la taula dels test de Bayes 1 podem observar que per a l'experiment 2 l'aproximació probabilística diu que els dos mètodes són igual de bons el 78'4% de les vegades i que aquests dos són millors que els altres mètodes. Així podem dir que els resultats per a l'experiment 2 han estat coherents i esperables
Finalment per al ***Experiment 3*** podem veure com l'algorisme no supervisat ha estat el millor de tots. No obstant això, aquest cop els diferents resultats han estat més propers entre ells. Si mirem la taula esmentada1 podem veure que en aquest cas la majoria d'algorismes es comporten igual probabilísticament excepte amb la que és pitjor. En la taula comparativa podem veure com realment aquest algorisme ha estat el pitjor mentre que els altres sí que són una mica més semblants i tenen un comportament coherent respecte a la taula de la comparativa de Bayes.

6 CONCLUSIONS

En aquest treball ha estat introduïda una bona comparativa per a diferents mètodes de Record Linkage. Malgrat els resultats i inconvenients en els mètodes del motor del neo4j s'ha elaborat una anàlisi rigorós i metòdic sobre els diferents experiments i les diferents branques experimentals.
Queda de manera òbvia el fet que s'han de fer experiments més exhaustius i de millor qualitat per aconseguir un rendiment desitjable per als algorismes del motor del Neo4j. Queda conclòs que per a problemes tan complicats i amb tal volum de dades utilitzar la versió comunitària de la BBDD no és òptim, almenys, en aquest cas.
En la primera branca experimental s'ha vist que afegir característiques sí que provoca una millora dels resultats mentre que en la segona branca experimental no s'ha pogut determinar estadísticament si és així.
En aquest treball podem concloure, a falta d'experiments més rigorosos com ha estat comentat, que els algorismes de machine learning i adaptacions han estat millors per encarar aquest problema de record linkage amb aquesta quantitat i disposició de les dades.
Ens trobem enfront d'un problema complicat amb una gran branca d'investigació. Aquest treball necessita una major anàlisi i diferents enfocaments per a poder encara'l de di-

		Experiment 1		Experiment 2		Experiment 3		Experiment 4		Experiment 5	
		F-score	Auc	F-Score	Auc	F-Score	Auc	F-Score	Auc	F-Score	Auc
Graph Sage	Cosine	0.467	0.717	0.350	**0.721**	0.385	**0.718**	0.212	0.651	0.152	0.612
	Minkowski	0.467	**0.717**	0.271	0.662	0.395	**0.718**	0.217	0.651	0.172	0.638
	Euclidean	0.460	0.7075	0.350	**0.721**	**0.401**	0.732	**0.243**	**0.671**	0.167	**0.643**
	Shannon-Jensen	0.301	0.701	0.300	0.671	0.388	0.716	0.183	0.648	0.156	0.625
NodeSimilarity	Jaccard	**0.738**	0.679	**0.750**	0.691	**0.689**	0.659	**0.712**	0.618	**0.578**	0.598

TAULA 9: LA TAULA MOSTRA ELS DIFERENTS RESULTATS AMB LES DIFERENTS FUNCIONS DE SIMILITUD. PER AL NODESIMILARITY NO HA ESTAT UTILITZADA LA FUNCIÓ OVERLAP PERQUÈ ENCARA ES TROBA EN FASE BETA DINTRE DEL NEO4J. EN NEGRE ESTAN MARCATS ELS MILLORS RESULTATS DE CADA EXPERIMENT PER CADA TIPUS DE MÈTRICA

ferents maneres i poder realitzar més comparacions. Caldria veure si amb la disposició de totes les eines que ofereix Neo4j els resultats podrien ser diferents.

AGRAÏMENTS

Aquest projecte ha estat suportat pel Centre de Visió per Computador, qui ha donat les dades, el Centre de Demografia Històrica de Catalunya, qui ha aportat la metodologia experimental, i pel meu tutor el Dr. Oriol Ramos Terrades qui ha aportat tot el coneixement necessari perquè pogués dur endavant aquest projecte.

REFERÈNCIES

[1] Josephine Sarpong Akosa. "Predictive Accuracy : A Misleading Performance Measure for Highly Imbalanced Data". A: 2017.

[2] A. Bordes et al. "Translating Embeddings for Modeling Multi-relational Data". A: *Advances in Neural Information Processing Systems 26*. Ed. de C. J. C. Burges et al. 2013, pàg. 2787 - 2795.

[3] A. I. Cowen-Rivers et al. "Neural Variational Inference For Estimating Uncertainty in Knowledge Graph Embeddings". A: *arXiv preprint arXiv:1906.04985* (2019).

[4] A. K. Elmagarmid, P. G. Ipeirotis i V. S. Verykios. "Duplicate Record Detection: A Survey". A: *IEEE Transactions on Knowledge & Data Engineering* 19 (gen. de 2007), pàg. 1 - 16. ISSN: 1041-4347. DOI: 10.1109/TKDE.2007.9.

[5] I. P. Fellegi i A. B. Sunter. "A Theory for Record Linkage". A: *Journal of the American Statistical Association* 64.328 (1969), pàg. 1183 - 1210. DOI: 10.1080/01621459.1969.10501049.

[6] Ivan P. Fellegi i Alan B. Sunter. "A Theory for Record Linkage". A: *Journal of the American Statistical Association* 64.328 (1969), pàg. 1183 - 1210. DOI: 10.1080/01621459.1969.10501049. eprint: https://www.tandfonline.com/doi/pdf/10.1080/01621459.1969.10501049. URL: https://www.tandfonline.com/doi/abs/10.1080/01621459.1969.10501049.

[7] Aditya Grover i Jure Leskovec. "node2vec: Scalable feature learning for networks". A: *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. 2016, pàg. 855 - 864.

[8] William L. Hamilton, Rex Ying i Jure Leskovec. "Inductive Representation Learning on Large Graphs". A: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. NIPS'17. Long Beach, California, USA: Curran Associates Inc., 2017, pàg. 1025 - 1035. ISBN: 9781510860964.

[9] Matthew A. Jaro. "Probabilistic linkage of large public health data files." A: *Statistics in medicine* 14 5-7 (1995), pàg. 491 - 8.

[10] J M. Pujades-More et al. "Chapter 2. The Baix Llobregat (BALL) Demographic Database, between Historical Demography and Computer Vision (nineteenth–twentieth centuries)". A: gen. de 2019. Cap. 2, pàg. 29 - 61. ISBN: 9785799626563. DOI: 10.15826/B978-5-7996-2656-3.03.

[11] Alvaro Monge i Charles Elkan. "An Efficient Domain-Independent Algorithm for Detecting Approximately Duplicate Database Records". A: (ag. de 2001).

[12] Vijayshankar Raman i Joseph M. Hellerstein. "Potter's Wheel: An Interactive Data Cleaning System". A: *VLDB*. 2001.

[13] Pradeep Ravikumar i William Cohen. *A Hierarchical Graphical Model for Record Linkage*. 2012. DOI: 10.48550/ARXIV.1207.4180. URL: https://arxiv.org/abs/1207.4180.

[14] Zhen Wang et al. "Knowledge Graph Embedding by Translating on Hyperplanes." A: *AAAI*. 2014, pàg. 1112 - 1119.

[15] William E. Winkler. *The State of Record Linkage and Current Research Problems*. Inf. tèc. Statistical Research Report Series RR99/04. U.S. Bureau of the Census, Washington, D.C., 1999.