Comparació de Descripcions Textuals i Formals de Models de Processos

Treball de fi de Grau

Enginyeria Informàtica

Alumne: Josep Sánchez Ferreres
Director: Josep Carmona Vargas
Co-Director: Lluís Padró Cirera

21 de Juny de 2016

Abstract

Business Process Management is a field in operations management that focuses on the management and optimisation of business processes in a company. Those business processes are often documented using both formal diagram languages and informal textual descriptions and inconsistencies between both often happen. The aim of this project is to develop an algorithm that, by using Natural Language Processing techniques, can provide a similarity metric between a textual description and a formal description in the BPMN notation of a business process.

The developed algorithm is capable of calculating a normalized similarity score based on the linguistic and structural features of both the formal model and the textual description. The algorithm has proven to be able to correctly match heterogeneous model and text pairs with high confidence.

Resum

El *Business Process Management* és un camp en l'administració de la producció que té per objectiu la gestió i optimització dels processos de negoci en una empresa. Aquests processos de negoci moltes vegades es documenten utilitzant llenguatges formals en forma de diagrama i descripcions textuals informals al mateix temps. En aquest escenari, és habitual l'aparició d'inconsistències entre les dues representacions. L'objectiu d'aquest projecte és desenvolupar un algorisme que, utilitzant tècniques de *Natural Language Processing*, pugui obtenir una mesura de similitud entre una descripció textual i una formal en la notació BPMN d'un procés de negoci.

L'algorisme desenvolupat és capaç de calcular una puntuació de similitud normalitzada basada en les característiques linguístiques i estructurals del model formal i la descripció textual. S'ha demostrat que l'algorisme pot agrupar parelles heterogènies de textos i models amb un alt nivell de confiança.

Resumen

El *Business Process Management* es un campo dentro de la administración de la producción que tiene como objetivo la gestión y optimización de los procesos de negocio en una empresa. Estos procesos de negocio muchas veces se documentan utilitzando, a la vez, lenguages formales en forma de diagrama y descripciones textuales informales. En un escenario como éste, es habitual la aparición de inconsistencias entre las dos representaciones. El objetivo de este proyecto es desarrollar un algoritmo que, utilizando técnicas de *Natural Language Processing*, pueda obtener una medida de similitud entre una descripción textual y una formal en la notación BPMN de un proceso de negocio.

El algoritmo desarrollado puede calcular una puntuación de similitud normalizada basada tanto en las características linguísticas como estructurales del texto y el modelo. Se ha demostrado que el algoritmo puede agrupar parejas heterogenias de textos i modelos con un alto nivel de confianza.

Agraïments

En aquest treball de final de grau hi han participat altres persones de manera directa o indirecta. Ja sigui guiant-me en el procés de desenvolupament, donant la seva opinió, oferint idees o simplement donant suport moral i acompanyant en els moments difícils. Vull agrair el seu ajut a tots els qui han contribuït a dur a terme aquest projecte.

En primer lloc, vull donar les gràcies als meus directors de TFG, Josep Carmona i Lluís Padró. D'una banda per guiar-me durant el llarg procés que ha estat aquest TFG, respondre sempre tots els meus dubtes i inquietuds i ser comprensius sempre que he tingut algun problema. Per altra banda, per la gran oportunitat que m'han ofert de seguir treballant amb ells després d'aquest treball de fi de grau. Estic segur que encara em queden moltes coses per aprendre de la seva experiència i coneixement.

També vull donar les gràcies a la Facultat d'Informàtica de Barcelona, i tots els seus professors, per uns quatre anys de carrera que m'han fet créixer tant acadèmica com personalment.

Per descomptat, gràcies també a tots els meus companys i amics Amb ells sempre he pogut compartir tant els riures com la motivació i el plaer pel coneixement. I gràcies molt especialment a la Montse, que ha aguantat les meves llargues explicacions tècniques, m'ha aconsellat, m'ha recolzat en els moments difícils i ha celebrat amb mi els moments d'alegria que m'ha aportat aquest projecte.

I no puc oblidar el meu reconeixement al paper que han tingut els meus pares. Sense ells, el seu suport, la seva dedicació a mi i la seva comprensió en els moments complicats, res d'això hagués estat possible. També agraeixo als altres familiars, tant pels ànims i la confiança que m'han donat com per l'interès que sempre han mostrat en els meus projectes.

A tots els qui m'han recolzat, moltes gràcies.

Índex

Capítol 1

Introducció

Un procés de negoci es defineix com una col·lecció de tasques descrites de forma estructurada que tenen per objectiu la producció d'un producte o servei concret. El *Business Process Management* és un camp en l'administració de la producció que té per objectiu la gestió i optimització dels processos de negoci en una empresa. Aquestes optimitzacions, sovint suposen una millora tant econòmica com en qualitat de vida dels treballadors. És per aquest motiu que cada vegada més empreses opten per aplicar tècniques de *Business Process Management*.

En el *Business Process Management* la documentació dels processos de negoci és la clau de l'èxit. Cal una forma estandarditzada, expressiva i inambigua de documentar els processos. Una forma de documentació molt utilitzada és una descripció textual en llenguatge natural. Com a alternativa, existeixen diverses notacions formals. El principal avantatge d'aquest tipus de notacions és no comptar amb les ambigüitats inherents al llenguatge natural. Una de les notacions més utilitzades és el *Business Process Model and Notation* (BPMN). No obstant això, el BPMN no és la manera preferida de descriure processos de negoci per a molts *stakeholders*. El resultat d'això és que en molts casos s'opta per documentar els processos tant en llenguatge natural com en BPMN. En un escenari com aquest, és habitual que apareguin inconsistències entre el model BPMN i el text, sobretot després de diverses actualitzacions sobre el procés de negoci en qüestió.

Aquest projecte busca resoldre precisament el problema plantejat. L'objectiu és desenvolupar un algorisme que permeti establir una mesura de similitud entre un model BPMN i una descripció textual amb la finalitat de detectar inconsistències entre els dos tipus de documentació. Per a fer-ho, es farà ús extensiu de tècniques tant de *Natural Language Processing* com de *Business Process Management*.

Aquesta memòria es troba repartida en 6 parts. El capítol 2 primer posa el projecte en el context dels diferents camps d'estudi que aquest engloba i, a continuació, proporciona una definició formal del problema que es resol. El capítol 3 descriu l'estratègia emprada per a resoldre el problema tot formalitzant matemàticament els conceptes que hi intervenen. A continuació, al capítol 4 es descriuen les parts més importants de la implementació del projecte, justificant les decisions preses i descrivint totes les alternatives estudiades. Seguidament el capítol 5 conté diferents experiments que pretenen avaluar la qualitat de l'algorisme de forma empírica. Finalment, al capítol 6 s'explica tot allò referent a com s'ha gestionat aquest projecte. Entre altres coses s'hi inclouen la planificació temporal, el pressupost i l'informe de sostenibilitat del projecte.

Capítol 2

Marc Teòric i Definició del Problema

2.1 Contextualització

Abans de definir formalment el problema a resoldre, cal posar aquest projecte en context parlant dels diferents camps d'estudi que hi intervenen: El *Business Process Management* i el *Natural Language Processing*.

2.1.1 Business Process Management

En el món corporatiu, a l'hora de descriure com opera una empresa se sol parlar en termes de processos de negoci (*business process*). Un procés de negoci es pot definir, de forma abstracta, com una col·lecció de tasques descrites de forma estructurada que tenen per objectiu la producció d'un producte o servei concret. En altres paraules, un procés és la descripció de com es duu a terme una certa operació en una empresa, indicant com es mou la informació a través dels diferents agents implicats i quines tasques es realitzen a cada pas.

És en aquest àmbit que apareix el camp del *Business Process Management* (BPM). El BPM engloba l'estudi i modificació d'aquests processos amb l'enfoc específic de millorar-los optimitzant com, i en quin ordre, es porten a terme les tasques que descriu.

Moltes empreses fan servir BPM tant per documentar com per millorar l'eficiència dels seus processos de negoci, i com a resultat d'això, mantenen grans repositoris d'informació amb aquests processos. És vital, doncs, mantenir aquesta informació actualitzada i coherent, i és en aquest problema on neix la motivació per aquest treball.

2.1.2 Natural Language Processing

El *Natural Language Processing* (NLP) és un camp en la informàtica que tracta les interaccions entre els ordinadors i el llenguatge natural dels humans. El NLP tracta diversos tipus de problemes a diferents nivells d'abstracció, des dels més purament lingüístics com l'anàlisi morfosintàctic d'oracions o determinar la categoria gramatical d'una paraula (*POS-tagging*) fins a la creació de resums automàtica de notícies d'un

diari.

Aquells problemes que tenen a veure, no només amb reconèixer i dividir el text sinó que també pretenen entendre'n el contingut i derivar-ne informació semàntica, cauen en el camp del *Natural Language Understanding* (NLU). L'objectiu del NLU és construir representacions semàntiques completes dels textos, de forma que siguin processables, i.e. comprensibles, per a una màquina. Es tracta, doncs, d'un problema IA-complert [22, secció 1], ja que els textos estan escrits per a lectors humans pressuposant sentit comú i coneixements del món: Coses molt difícils d'incloure en un programa.

2.1.3 Descripcions textuals i models de processos

És evident la necessitat de representar d'alguna manera els processos de negoci per tal que els actors implicats puguin entendre'ls així com perquè els encarregats del *business process management* puguin millorar-los.

La primera forma de representar processos que es tractarà en aquest treball, i també la més evident, és la textual. Consisteix en descriure el procés en un document fent servir llenguatge natural. La figura 2.2 mostra un exemple de representació textual d'un procés. Tot i generalment ser la més senzilla d'interpretar [12] per les persones implicades en el procés, el fet que es tracti d'informació no estructurada i la ambigüitat inherent al llenguatge natural dificulten molt el tractament d'aquests processos des de l'enfoc del BPM. És per això que apareix el *business process model and notation* (BPMN).

El BPMN és un estàndard de representació gràfica de processos de negoci creat per la *Business Process Management Initiative* (BPMI)[1]. La figura 2.1 mostra un exemple d'aquesta notació. L'apartat 2.1.4 conté una descripció més detallada del BPMN.

El BPMN ofereix nombrosos avantatges sobre la representació textual: En primer lloc, permet representar de forma molt més inambigua el contingut, tractant-se d'informació estructurada en forma de diagrama. A més, donada la seva estructura es pot tractar molt més fàcilment amb software; permetent aplicar anàlisi formal per a extreure conclusions globals sobre el comportament dels processos. Finalment, existeixen motors d'execució que a partir d'un BPMN controlen l'execució d'un procés a temps real, permetent a més executar automàticament totes les tasques que puguin ser definides per un *script*. Tot això fa que l'ús del BPMN sigui molt valuós per a una empresa que vulgui automatitzar l'anàlisi i execució dels seus processos.

Els avantatges tecnològics del BPMN respecte a la representació textual són evidents. No obstant això, diferents tipus d'*stakeholders* prefereixen diferents tipus de notació perquè els hi resulta més fàcil d'entendre o perquè no coneixen bé el BPMN.

Representació dual

Com que no hi ha un clar vencedor entre la representació textual i la notació BPMN a l'hora de representar processos, moltes vegades s'opta per mantenir el procés documentat d'ambdues maneres. Tenint en compte que en una empresa gran s'estan documentant centenars d'aquests processos, mantinguts per diversos grups de persones i utilitzant diferents eines d'edició és molt probable que tard o d'hora es produeixin inconsistències entre la descripció textual i el model BPMN que representen el mateix procés. Aquestes incoherències són una font de problemes en el procés de producció de l'empresa en qüestió, i solucionar-los manualment requereix una gran inversió en temps i diners.

[1] Actualment l'estàndard de BPMN està mantingut per l'*Object Management Group* (OMG).

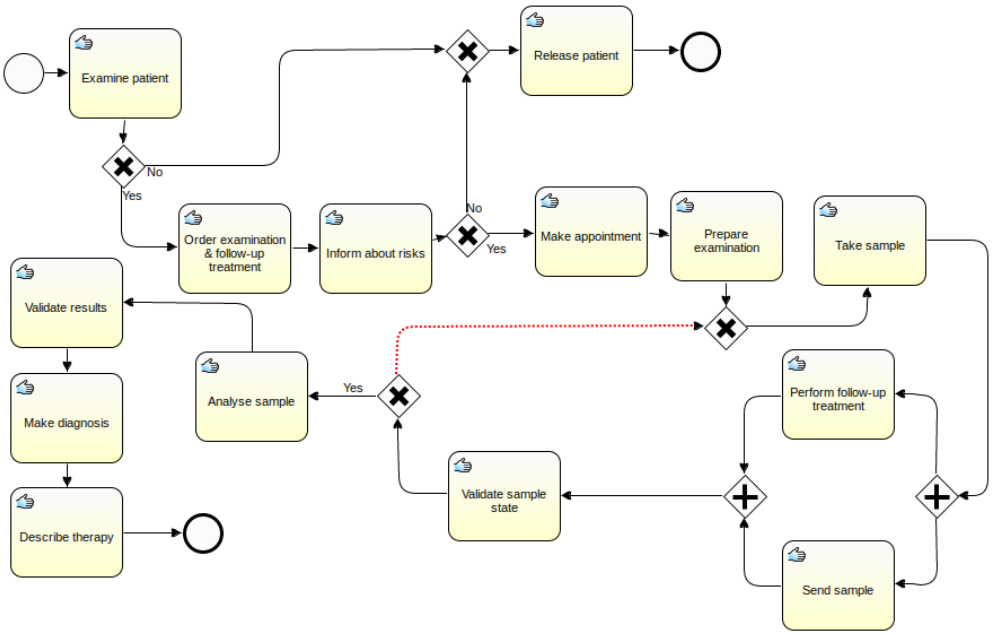


Figura 2.1: Exemple de model BPMN corresponent al text de la figura 2.2

The examination process can be summarised as follows. The process starts when the female patient is examined by an outpatient physician, who decides whether she is healthy or needs to undertake an additional examination. In the former case, the physician fills out the examination form and the patient can leave. In the latter case, an examination and follow-up treatment order is placed by the physician who additionally fills out a request form. Beyond information about the patient, the request form includes details about the examination requested and refers to a suitable lab. Furthermore, the outpatient physician informs the patient about potential risks. If the patient signs an informed consent and agrees to continue with the procedure, a delegate of the physician arranges an appointment of the patient with one of the wards. The latter is then responsible for taking a sample to be analysed in the lab later. Before the appointment, the required examination and sampling is prepared by a nurse of the ward based on the information provided by the outpatient section. Then, a ward physician takes the sample requested. He further sends it to the lab indicated in the request form and conducts the follow-up treatment of the patient. After receiving the sample, a physician of the lab validates its state and decides whether the sample can be used for analysis or whether it is contaminated and a new sample is required. After the analysis is performed by a medical technical assistant of the lab, a lab physician validates the results. Finally, a physician from the outpatient department makes the diagnosis and prescribes the therapy for the patient.

Figura 2.2: Exemple de text que descriu un procés de negoci.

Donat aquest problema, és evident el gran avantatge d'automatitzar la tasca de comparar el model BPMN amb la corresponent descripció textual, però la dificultat per tractar amb textos no estructurats fa que aquest sigui un terreny molt inexplorat.

2.1.4 La notació BPMN

En els apartats anteriors s'ha parlat de la notació BPMN. La funció de BPMN és modelar processos de negoci (Veure apartat 2.1.3) i és un estàndard àmpliament reconegut i utilitzat per moltes empreses. A nivell d'implementació, està definit com un subconjunt (schema) de XML. L'especificació també descriu una notació visual que s'utilitza per editar i visualitzar els models de procés. En aquesta secció es fa només una breu explicació de la notació BPMN. No obstant, l'especificació d'aquest llenguatge és molt àmplia i, per tant, no es pretén donar una explicació exhaustiva. A [11] se'n pot consultar una especificació completa.

El més important de la notació BPMN respecte altres alternatives és que defineix una semàntica d'execució. És a dir, donat un model BPMN, l'execució d'aquest es pot simular de manera automàtica i inambigua. Això permet que les eines de BPM que l'utilitzen puguin coordinar i fins i tot executar algunes parts del procés que defineixen.

En els següents apartats es defineix cadascun dels elements que formen part d'un model BPMN.

Tasques

El primer element, i concepte central als processos de negoci, són les tasques. Una tasca és una acció que s'ha de realitzar en algun punt en l'execució d'un procés de negoci. Les tasques es representen amb un rectangle amb text a dins. El text ha de contenir simplement el nom de la tasca. Altra informació com el responsable de realitzar-la, o execució sota una condició, s'ha d'indicar mitjançant altres construccions. La figura 2.3 mostra un exemple d'una tasca.

Figura 2.3: Una tasca

La granularitat de les tasques depèn de l'àmbit de cada problema. En alguns casos caldrà considerar les tasques a un nivell més baix i en altres casos una tasca pot representar diverses hores. La granularitat correcta per a un procés concret és la més gran possible, sempre i quan les tasques segueixin sent atòmiques[2]. Per exemple, suposem que el cas d'un procés d'un hospital per a determinar si un malalt pateix una certa malaltia. En aquest procés hi podríem trobar tasques com "Examinar el pacient" o "Analitzar les mostres". Tot i això, en un procés del mateix hospital on s'està parlant

[2]En altres paraules, no es pot indicar lògica en mig d'una tasca, però no té sentit indicar tots i cadascun dels passos d'una tasca si s'han de realitzar seqüencialment i sense interrupcions.

de com analitzar els símptomes d'un pacient durant l'examinació, la granularitat de la tasca "Examinar Pacient" és massa gran.

Sequence Flows

El sequence flow és l'element bàsic per indicar el control de flux entre les tasques. Un sequence flow connecta tasques, events i gateways entre ells. És l'equivalent a una aresta en un graf dirigit i es representa, de forma similar, amb una fletxa (Figura 2.4). Degut a aquesta similitud, a la literatura s'utilitza moltes vegades vocabulari similar al dels grafs, anomenant-los arestes o arcs.

Figura 2.4: Diferents tasques unides per sequence flows.

Un sequence flow indica una dependència entre dues tasques. Si la tasca A i la tasca B estan connectades per un sequence flow, la tasca A s'ha de realitzar estrictament abans que B.

Events

Els events serveixen per modelar successos que poden ocórrer durant l'execució d'un procés. Exemples d'events són: "Es rep una trucada", "Acaba l'execució del programa" o "Cada 10 minuts". Els events es representen amb un cercle. La figura 2.5 mostra diferents tipus d'events. Els events es poden connectar a altres events, o tasques, mitjançant sequence flows.

Existeixen tres tipus d'events: Start Event, Intermediate Event i End event. Un Start Event és aquell que no té arcs d'entrada. L'execució del model comença sempre en un Start Event. D'altra banda, els End Events sempre acaben un procés, i per tant no poden tenir arcs de sortida. Finalment, els Intermediate Events serveixen per modelar events que puguin ocórrer durant l'execució d'un procés. La tasca que genera un event intermig està connectada a aquest. En un altre punt del model, el mateix event pot tenir un arc de sortida cap una tasca, indicant que quan es produeixi l'event es realitzarà la tasca en qüestió.

Gateways

Les gateway serveixen per indicar el control de flux en un model de procés de negoci. Amb les gateway es poden modelar les construccions típiques dels llenguatges de programació: Execució condicional i Execució en paral·lel. Es representen gràficament amb un quadrat rotat 90 graus. La figura 2.6 en mostra els diferents tipus.

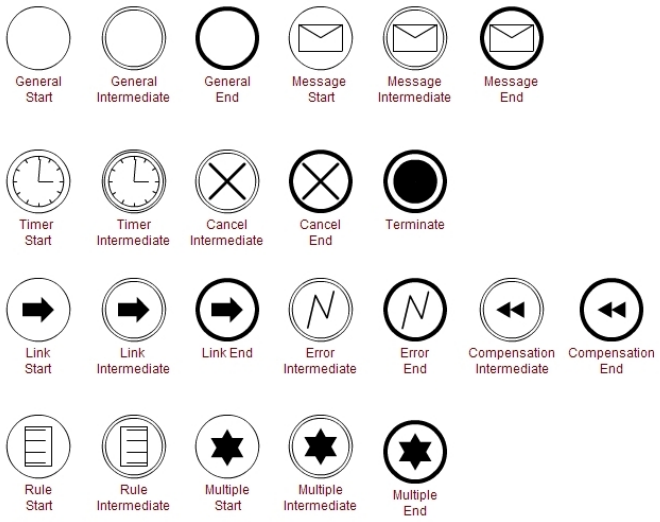


Figura 2.5: Diferents tipus d'events.

Figura 2.6: Diferents tipus de gateway, d'esquerra a dreta: Basic, Exclusive, Parallel, Event-Based, Complex i Inclusive.

Existeixen diferents tipus de gateway, cadascuna amb una semàntica d'execució associada. Les mes típiques són la Exclusive gateway i la Parallel gateway. La exclusiva indica una ramificació condicional de l'execució en diversos camins, dels quals s'agafarà un (i només un) en funció de la resposta a una certa pregunta. La paral·lela indica també una ramificació de l'execució, però en aquest cas s'agafen tots els camins i s'executen en paral·lel. Els altres tipus de gateways, menys utilitzats, són la Inclusive Gateway, la Event-Based Gateway i la Complex Gateway, cadascuna amb la seva semàntica particular d'execució. No obstant, és típic a la literatura fer una simplificació i considerar només gateway de tipus paral·lel i exclusiu, ja que les altres es poden modelar a partir d'aquestes dues.

Pools i swimlanes

En el text d'una tasca s'hi descriu típicament l'acció, però la frase que la descriu mai acostuma incloure el subjecte. Això és així perquè aquesta és la tasca de les pools i les swimlanes. Les pools i swimlanes són contenidors dels elements anteriorment mencionats: Tasques, sequence flows, events i gateways. Tant les pools com les gateways tenen un nom, i el nom indica qui, o què, realitza les accions que aquest conté. La figura 2.7 conté un exemple de procés amb una pool i diverses swimlanes.

La diferència entre les pools i les swimlanes és jeràrquica. Una pool pot contenir una o varies swimlanes. A nivell semàntic, la pool sol indicar l'organització que realitza l'acció mentre que la swimlane es refereix a l'actor concret. Per exemple, el director del departament de Recursos Humans, estaria dins la swimlane "Director" continguda dins

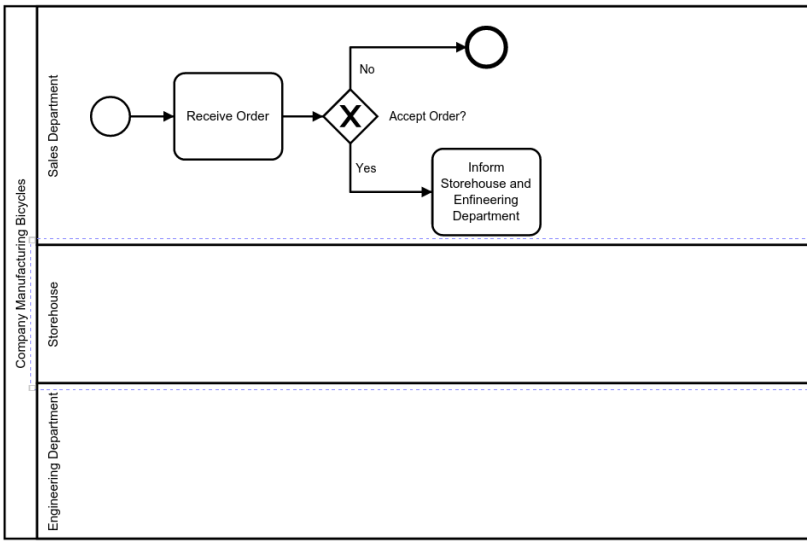


Figura 2.7: Una pool amb tres swimlanes.

la pool "Recursos Humans".

No pot existir cap sequence flow entre elements de diferents pools, però si entre elements de diferents swimlanes. Generalment es diu que cada pool conté un *graf de procés* diferent. Per modelar la comunicació entre elements de diferents pools, s'utilitzen els message flows.

Message flows

Els message flow serveixen per indicar la comunicació entre elements de diferents pools. Per exemple, un message flow pot enllaçar una tasca d'una pool amb un event d'una altra pool, indicant que quan aquella tasca es realitzi, s'activarà l'event en una altra pool. Els message flow es representen amb fletxes igual que els sequence flow però amb una línia discontinua.

2.2 Estat de l'art

Tant el *Business Process Management* com el *Natural Language Processing* són dues disciplines ben establertes en el camp de la informàtica. Tot i això, l'aplicació de tècniques de NLP per a BPM és un camp d'estudi molt recent. En aquesta secció es discuteix l'estat de l'art en les dues disciplines per separat i també com a disciplina conjunta.

2.2.1 Business Process Management

El Business Process Management ha rebut molta atenció durant els darrers anys degut al seu potencial per millorar la productivitat de les empreses. En el camp, trobem principalment dues vessants. La primera, que consisteix en l'ús dels models de processos com un suport a la presa de decisions en un entorn corporatiu, permetent als diferents *stakeholders* entendre el problema en qüestió. La segona vessant –més interessant des

del punt de vista de la informàtica-, està encarada en l'automatització dels processos per a permetre'n la coordinació automàtica. Aquest darrer camp és conegut pel nom de *Business Process Automation*. Gràcies a les tècniques desenvolupades en BPA, és possible utilitzar eines de software que controlin els processos, executant les tasques automàticament quan això sigui possible, i entregant-les al recurs pertinent, persona o una altra màquina, en cas contrari. Com a producte d'aquesta coordinació automàtica, s'obtenen diverses traces d'execució amb informació detallada que en permeten un anàlisi posterior. Per exemple, si observant les traces es detecta que una tasca concreta és el *bottleneck* del procés, es pot estudiar una solució al problema per millorar la productivitat. Aquest tipus d'anàlisi també juga un paper molt important, no només en l'optimització, sinó també en la detecció d'errors. El conjunt de tècniques d'anàlisi de les traces d'execució rep el nom de *Process Mining*.

Un dels obstacles més grans en el camp del BPM actualment és la falta de consens en la formalització dels processos. Existeixen diversos estàndards per modelar processos de negoci, i no tots estan enfocats en l'automatització dels processos que modelen. Aquesta falta de consens es pot atribuir principalment a dos fets:

1. La complexitat inherent en molts processos de negoci. Ja que moltes vegades les diferents notacions per modelar processos no permeten expressar segons quin tipus de regles complexes.
2. Els diferents intents en el camp d'estandarditzar una única notació formal per al business process management no han funcionat, creant tot tipus de potencials estàndards diferents.

Tot i que l'enfoc principal en el camp se centra en la millora de productivitat dels processos, diversos estudis destaquen la importància de la verificació formal de processos. La detecció de possibles contradiccions lògiques en un procés de negoci és crucial, sobretot per a permetre'n l'automatització. Si no es verifiquen els processos, poden aparèixer errors com *deadlocks* o *livelocks* en l'execució d'un procés. A la literatura, existeixen mesures com la *soundness*[3] que pretenen analitzar la correctesa dels models.

Recentment, s'ha començat a investigar la utilització de tècniques de processat de llenguatge com a eines per al *Business Process Management*. La motivació és, per una banda facilitar el modelat dels processos, i per altra banda permetre l'anàlisi i formalització automàtica de processos no especificats en un llenguatge formal. Han aparegut diversos estudis busquen superar alguns dels obstacles descrits del camp del BPM utilitzant NLP. Alguns d'aquests són:

- La conversió automàtica de descripcions textuals, en llenguatge natural, a models de processos [10].
- La conversió automàtica de models de processos a documents en llenguatge natural.
- La comparació automàtica de descripcions textuals i models de processos, camp en el que es troba emmarcat aquest projecte [1].

Un dels desenvolupaments més recents en el processat de llenguatge natural per a BPM és un estudi[2] on es proposa el concepte de *behavioral space*. L'àmbit de l'estudi se centra en les descripcions textuals de processos, concretament en el fet que les tècniques de NLP no tenen la mateixa capacitat que un humà a l'hora de desambiguar una possible interpretació del significat de les accions d'un procés d'entre totes les possibles.

El behavioral space és un enfoc sistemàtic al problema de capturar totes les possibles interpretacions d'una descripció textual d'un procés, en comptes d'haver d'assumir-ne una com a correcta. Els recents resultats obtinguts poden suposar una nova eina molt útil a l'hora de modelar automàticament descripcions textuals de processos.

2.2.2 Natural Language Processing

En el camp de NLP el focus principal està en interpretar textos cada vegada més complexos, inferint coneixement implícit en el context. Els diferents problemes resolts, en major o menor mesura en NLP són[17]:

- Separació de textos en frases.
- Tokenització, o separació de frases en unitats més simples com paraules o signes de puntuació.
- Lematització i Stemming. És a dir, la reducció d'una paraula a la seva arrel, ja sigui utilitzant diccionaris com en el cas de la lemmatització, o aplicant regles simples com en el cas del stemming.
- POS-tagging. El procés de determinar la categoria gramatical de cada paraula[3].
- Parsing de constituents, que consisteix a determinar els diferents sintagmes en una frase.
- Parsing de dependències, que consisteix a determinar les funcions i relacions entre els elements de la frase. Correspon a l'anàlisi sintàctic.

Actualment, l'enfoc en el camp del NLP és sobretot el *Natural Language Understanding* (NLU). El NLU pretén interpretar textos en llenguatge natural per extreure'n informació a alt nivell. Alguns dels problemes encara no resolts en el camp del NLU són:

Traducció automàtica intel·ligent Consisteix, no només en traduir textos utilitzant el diccionari, sinó considerar el context de la traducció i coses com frases fetes a l'hora de traduir.

Correcció d'errors Consisteix en corregir el text, típicament escrit amb errors, amb possibles errors abans de processar-lo. Un exemple es el pre-tractament de missatges en xarxes socials.

Obtenció d'informació Similar a un cercador, pero acceptant preguntes formulades en llenguatge natural, i no amb keywords.

Extracció d'informació A partir d'un text en llenguatge natural, extreure'n les paraules més rellevants i/o fer un resum automàticament.

El processat de llenguatge natural ha fet servir clàssicament sistemes de regles deterministes com les gramàtiques incontextuals. Més recentment, però, s'han començat a utilitzar tècniques basades en el Machine Learning[21] per aprendre a partir de textos reals. Els recents avenços amb les tècniques de machine learning també han permès millores en el processat de llenguatge natural[7].

[3]No només *Determinant*, sinó la categoria completa com *Determinant possessiu*

2.2.3 Comparació de descripcions textuals i models formals de processos

El problema que tracta aquest projecte, la comparació de models textuals i models de processos en BPMN, és un camp molt inexplorat. Només hi ha un grup [1] que hagi publicat un article proposant una solució a aquest problema.

La solució que proposa [1] consisteix, igual que aquest projecte, en combinar el BPM amb tècniques de NLP. L'algorisme (veure figura 2.8, a grans trets, és el següent:

1. Separar el text en frases i el model en tasques.
2. Analitzar i simplificar el text del model i de les frases amb una eina de NLP[4].
3. Computar una *score* de similitud entre les parelles frase-tasca.
4. Trobar una assignació òptima que doni la correspondència entre frases del text i tasques del model.
5. A partir de la informació computada, trobar les inconsistències entre el model i el text.

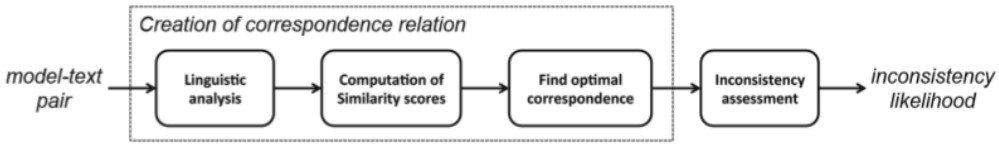


Figura 2.8: Estructura bàsica de la solució proposada en [1]

Centrant-nos en l'apartat d'anàlisi lingüístic, El tractament que es fa a les paraules del text és el següent:

Resolució d'anàfores: Els pronoms (*him*, *her*, ...), i alguns determinants (*this*, *that*, ...) del text se substitueixen pel mot al qual fan referència:
him → *manager*

Extracció de clàusules: S'eliminen les clàusules secundàries del text, per centrar-se només en les parts de la frase que defineixin una acció del model i eliminar mots que podrien afegir soroll a l'hora de computar similituds.

Sanitització del text: S'eliminen les *stopwords*[5] i les paraules restants se substitueixen per l'arrel del mot:
prepared → *prepare*

2.3 El problema

El problema que es vol resoldre amb aquest projecte es pot resumir amb el següent enunciat:

[4]Concretament l'eina utilitzada és l'*Stanford Parser* [14]
[5]Paraules que no aporten significat substancial a la frase, com articles o preposicions.

> "Donats un model BPMN i una representació textual d'un procés, determinar si corresponen al mateix procés donant una puntuació de similitud entre ells i informació per ajudar a detectar les possibles incoherències que hi pugui haver entre els dos."

L'objectiu principal del treball és, doncs, la creació de l'algorisme que resol el problema anterior. Més específicament, el treball s'ha dividit en els següents sub-objectius:

1. Utilitzar tècniques de NLP, fent servir *freeling*[6] des del *Textserver*[7], per extreure informació de la descripció textual d'un procés.
2. Explorar models BPMN i analitzar el seu contingut amb tècniques de NLP i Business Process Management per tal d'extreure'n informació.
3. Crear una representació homogènia per la informació extreta del model BPMN i la descripció textual.
4. Determinar una mètrica de similitud per comparar la informació extreta.
5. Establir una noció d'ordre entre elements del text i del model.
6. Utilitzar un algorisme de satisfacció de restriccions per calcular la correspondència de tasques a frases.
7. Recopilar tota la informació obtinguda per tal d'ajudar a detectar possibles incoherències entre el text i el model.

2.4 Abast

Un cop definit el problema que es vol resoldre, cal establir uns límits realistes i definir què es desenvoluparà exactament. En aquesta secció es defineix el producte mínim que es vol aconseguir, possibles ampliacions a aquest i els diferents obstacles als que el projecte s'enfronta.

2.4.1 Producte Esperat

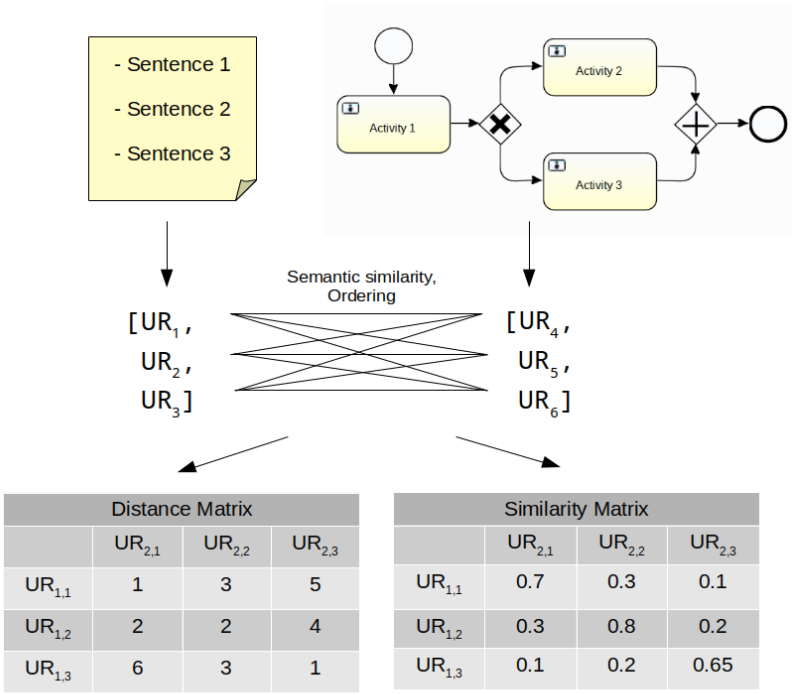
L'objectiu d'aquest projecte és dissenyar un algorisme que resolgui el problema descrit a la secció 2.3. El procés descrit per aquests objectius es pot visualitzar al diagrama de la figura 2.9. De les frases i el model se n'extreuen els *UR*[8] (Objectius 1, 2 i 3). A continuació es comparen els *UR* d'un cantó amb els de l'altre, per obtenir tant la matriu de similitud (Objectiu 4) com la de distància (Objectiu 5). Aquestes matrius passen a ser les dades d'entrada d'un algorisme d'optimització per establir l'assignació òptima[9] (Objectiu 6). Finalment de la resposta de l'algorisme se n'extreu la informació necessària per ajudar a detectar les possibles incoherències que hi puguin haver (Objectiu 7). Per una descripció àmplia de com s'ha plantejat abordar el problema i cadascun d'aquests objectius veure el capítol 3.

[6] *Freeling* [17] és una eina de software lliure per a *Natural Language Processing* creada pel professor Lluís Padró de la Universitat Politècnica de Catalunya.

[7] El *Textserver* [18] és una plataforma online que proporciona serveis web d'anàlisi lingüístic de textos. Fa servir freeling.

[8] Abreviació de *Unified Representation*

[9] Cal notar que aquesta assignació no ha de ser necessàriament bijectiva.



Distance Matrix			
	$UR_{2,1}$	$UR_{2,2}$	$UR_{2,3}$
$UR_{1,1}$	1	3	5
$UR_{1,2}$	2	2	4
$UR_{1,3}$	6	3	1

Similarity Matrix			
	$UR_{2,1}$	$UR_{2,2}$	$UR_{2,3}$
$UR_{1,1}$	0.7	0.3	0.1
$UR_{1,2}$	0.3	0.8	0.2
$UR_{1,3}$	0.1	0.2	0.65

Figura 2.9: Diagrama que representa l'estructura general de l'algorisme.

Com a possibles ampliacions al projecte base, s'han plantejat els següents objectius addicionals:

1. Estudiar les possibilitats de reduir el problema a un LAP[10], que té solució en temps polinòmic, per determinar una solució inicial per l'algorisme d'optimització.
2. Experimentar quina és el millor subconjunt d'informació a extreure eliminant possibles causes de soroll al càlcul de similitud.
3. Experimentar amb diverses mètriques de similitud i fer una comparativa per determinar quina s'escau més al domini del problema.
4. Investigar i experimentar diferents tipus d'algorismes d'optimització per determinar quin és el més eficient i ofereix millors solucions al problema objectiu.
5. Estudiar quins tipus d'inconsistències es poden extreure a partir de la informació calculada.

2.4.2 Obstacles

Els obstacles als que s'enfronta aquest projecte provenen de diverses fonts. En alguns casos el problema és tenir massa alternatives i haver d'escollir la millor. En altres casos

[10] *Linear Assignment Problem* [16]

és la falta d'informació i/o alternatives. D'altres són dificultats tècniques que queden totalment fora de l'abast del projecte. Aquest apartat és un recull de les possibles limitacions i obstacles que han aparegut en plantejar el projecte:

- La falta de coneixement del món per part de les eines de NLP utilitzades dificulten la possibilitat de determinar que dues frases referint al mateix. Aquest problema es pot exemplificar amb les frases: *The patient can leave* (text) i *Release patient* (BPMN). Les dues frases corresponen a la mateixa acció, però l'elecció de paraules del segon implica un coneixement del món de la medicina mitjançant el qual es pot deduir que el pacient és una persona que ha estat donada d'alta i que per tant els verbs *release* i *leave* s'estan referint a la mateixa acció quan en general no ho fan.
- La falta de dades d'exemple, ja que les empreses no solen compartir la documentació dels seus processos per motius de secret professional. Sense un repositori de dades reals d'exemple és difícil experimentar amb diferents opcions de l'algorisme i avaluar-ne la qualitat. Aquesta falta de dades també dificulta possibles opcions com el machine learning a l'hora de determinar els paràmetres de l'algorisme.
- La quantitat d'opcions quant a algorismes d'optimització i mètriques de similitud fa que quedi fora de l'abast d'aquest projecte provar-los tots i avaluar-ne la qualitat.
- Donat que el projecte pretén comparar textos i models BPMN, és difícil avaluar la qualitat de l'algorisme de forma automàtica, ja que per fer-ho caldria resoldre el mateix problema que pretén resoldre aquest.

2.5 El projecte NLP4BPM

Aquest projecte es troba dins el marc d'un projecte més gran: NLP4BPM[11]. El projecte NLP4BPM és una iniciativa que pretén combinar els mons del *Business Process Management* i el *Natural Language Processing* amb l'objectiu de la creació d'una eina per assistir en l'automatització de tasques relacionades amb el camp del BPM. El projecte s'ha desenvolupat fins al moment a partir de diversos treballs de final de grau, un d'ells encarregat d'integrar tots els algorismes en una plataforma web que proporcioni una interfície gràfica uniforme per a tot el software. Les diferents parts que s'han implementat amb diversos treballs han estat:

- Un traductor de descripcions textuals a models BPMN. Desenvolupat per Dídac Martínez i Arnau Gil.
- Un traductor de models BPMN a descripcions textuals. Desenvolupat per Genís Martín.
- Una eina que permeti comparar models BPMN i descripcions textuals. Desenvolupada per Josep Sánchez Ferreres. Aquesta eina correspon al projecte d'aquesta memòria.
- Una eina que permeti comparar models BPMN entre ells. Desenvolupada per Luis Delicado.
- Una eina gràfica (figura 2.10) que integri totes les funcionalitats anteriors en una plataforma web. Desenvolupada per Luis Delicado.

[11] *Natural Language Processing* for *Business Process Management*

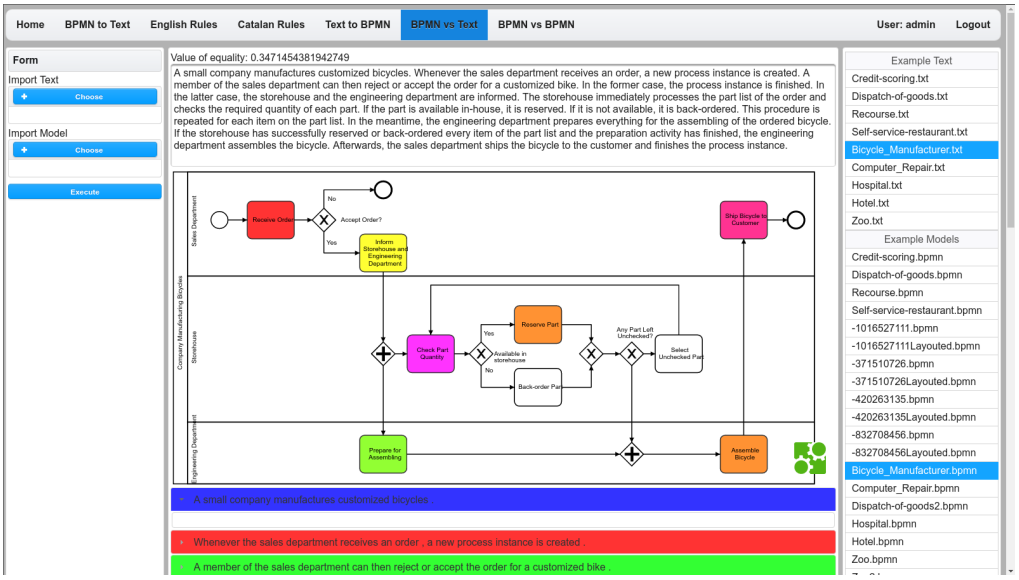


Figura 2.10: Captura de pantalla de la versió final del projecte executant-se a la plataforma web.

Capítol 3

Enfoc

3.1 Estructura general de l'algorisme

L'objectiu de l'algorisme es donar una mesura de similitud entre un model BPMN i la descripció textual d'un procés. Per fer-ho, es segueix l'estructura de la figura 3.1. L'objectiu és trobar quines frases del text i quines tasques del model estan relacionades per poder analitzar si aquesta relació presenta algun problema: Tasques del model que no es mencionen al text o viceversa, i casos en que l'ordre relatiu dels events no és el mateix.

A grans trets, la informació del text i del model es parteix en unitats més simples. Posteriorment, se'n fa una extracció de característiques creant una representació uniforme pels elements del text i els del model. Tot seguit s'estableix una mesura de similitud entre aquests elements. Paral·lelament, s'estableix un ordre parcial pels elements del text i els elements del model. Finalment tota aquesta informació –La similitud i l'ordre entre elements– es fa servir per calcular la correspondència òptima de tasques a frases. En els següents apartats s'explica formalment i amb més detall l'enfoc plantejat per a cadascuna d'aquestes parts.

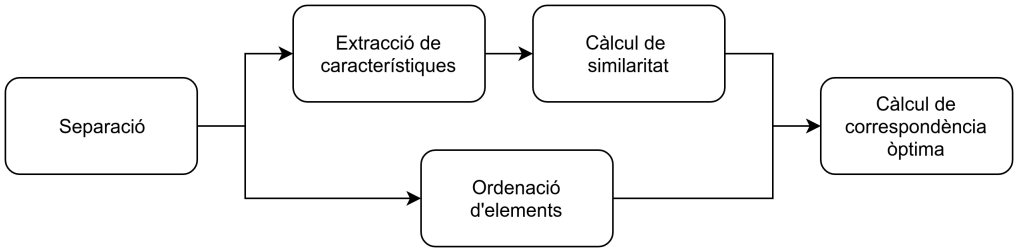


Figura 3.1: Esquema de l'estructura de l'algorisme.

3.2 Dades d'entrada

Per tal de comparar el model BPMN amb el text, cal partir aquests en elements més simples. S'ha escollit dividir el text en frases i el model BPMN en tasques.

Partir el text en tasques és l'alternativa més evident pel cas que ens plantegem, no

obstant això trobem moltes vegades casos on la frase no és la unitat òptima. Considerem la frase: "The chef prepares the meal and places it in the service hatch". Aquesta frase conté la descripció de dues accions clarament diferenciades, *"Prepare Meal"* i *"Place it (meal)"*. Aquesta partició més fina es correspondria a partir la frase en *predicats*, no obstant s'ha escollit simplificar el problema i considerar el text com una llista de frases.

D'altra banda, partir el model en tasques és una alternativa d'entrada contraintuïtiva. En un model BPMN les tasques juguen un paper molt important, però hi ha altres elements que aporten bona part del significat: Les *swimlanes* i *pools*, les *gateways* i el propi flux d'execució. De fet, l'enfoc no és considerar les tasques aïllades, sinó les tasques i el seu entorn. Així doncs, a l'hora de considerar la tasca *"Clarify Shipment Method"* en la figura 3.2, no només ens fixarem en el text de la tasca sinó que es considerarà també coses com que la *swimlane* conté el text *"secretary"*. Així evitem no perdre informació a l'hora de considerar el model.

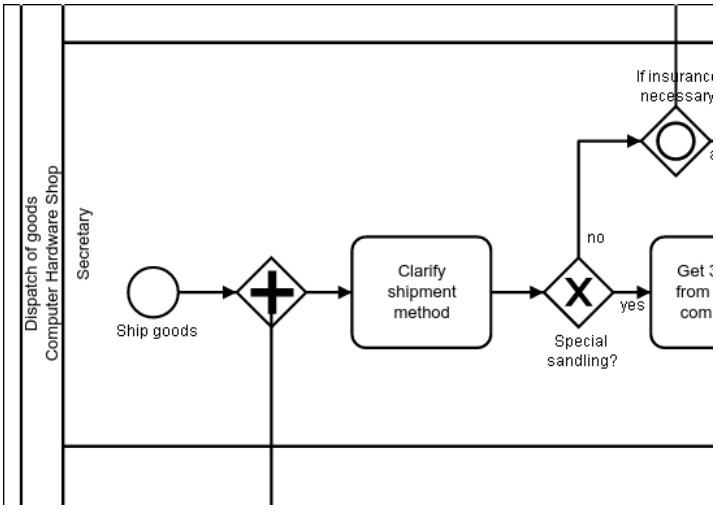


Figura 3.2: Exemple il·lustratiu d'un fragment de model BPMN

Així doncs, definim l'entrada de l'algorisme com un conjunt $S = \{s_1, ..., s_N\}$ de frases i un conjunt $T = \{t1, ..., tM\}$ de tasques. Definim informalment els elements dels conjunts seguint la definició que hem donat en aquest apartat.

3.3 Extracció de característiques

Com que les frases i les tasques son representacions d'informació totalment diferents, és difícil calcular directament la similitud entre aquestes. Si s'intentés calcular la similitud directament, els càlculs estarien plens de casos especials que barrejant informació molt concreta dels dos costats. És per això que s'ha afegit un pas d'extracció de característiques, convertint els diferents elements en una representació comuna tant per tasques com per frases: els vectors de característiques.

Per a aquest projecte s'ha escollit un espai molt gran, potencialment infinit, de característiques $F = f1, f2, ...$ de tipus binari. Cadascuna d'aquestes característiques representa la presència d'algun atribut distintiu de l'objecte al que representa. Per exemple, la frase *"The employee hands over his meal"* té la característica "El subjecte és employee", o en una notació més formal: $subjecte\_es(employee)$. Cal fixar-nos però,

que en un espai com aquest existeixen *tipus* de característiques. Definim, doncs, el conjunt: $\{f(x)|x \in \Omega^n\}$ com el tipus de característica de f. Així doncs el tipus de característica $subjecte\_es$ conté característiques concretes com $subjecte\_es(employee)$ i $subjecte\_es(secretary)$. Cal notar com el valor x d'una característica de tipus f és una tupla ordenada de n elements de qualsevol tipus. Aquest valor n l'anomenarem *aritat* del tipus de característica i depèn del tipus concret.

L'objectiu és representar les tasques i les frases com a vectors de característiques. A nivell teòric, un vector de característiques $v = [v_1, v_2, ..., v_n], v_i \in 0, 1$ té una dimensió per característica, amb un valor binari indicant si la característica és present en aquell objecte concret. Amb l'enfoc plantejat, n és potencialment infinita. Com que aquesta representació no és viable per al nostre espai de característiques, s'ha optat per representar el vector de característiques com un subconjunt de F de les característiques concretes que tindrien valor 1 al vector de característiques. Aquesta representació és típica a la literatura quan l'espai de característiques és infinit.

Definim finalment una funció extractora d'un tipus d'objecte X com una funció $f_X : X \to \mathcal{P}(F)$[1]. En el domini d'aquest problema, definim les funcions extractores de manera diferenciada per les tasques i les frases. Així doncs tenim funcions extractores f_{Task} i $f_{Sentence}$. Cal notar que els tipus de característiques que s'extreuen de les tasques i les frases són les mateixes, això és important a l'hora de comprar-les.

Formalment, l'extracció de característiques consisteix a convertir els conjunts S i T en els conjunts $S_{features}$ i $T_{features}$ de vectors de característiques. Definim $S_{features}$ com $\{f_{Sentence}(s)|s \in S\}$, i anàlogament pel cas de $T_{features}$.

3.4 Càlcul de similitud

Un cop hem reduït el problema de comparar frases i tasques a comparar vectors de característiques homogenis, calcular la similitud entre aquests és un problema ben conegut a la literatura[15]. Diferents tipus de dominis es beneficien més de diferents tipus de mètriques de similitud. En aquest apartat considerem algunes de les mètriques més utilitzades en aquest àmbit que s'han considerat útils per al problema en qüestió.

similitud de cosinus Considerant els vectors de característiques com a vectors en el sentit geomètric, amb una dimensió per característica, aquesta mètrica considera dos vectors més similars com més petit és l'angle entre ells dos. Es fa servir el fet que $u \cdot v = |u||v|cos(angle(u, v))$:

$$Cosine(A, B) = \frac{A}{|A|} \cdot \frac{B}{|B|}$$

Si els dos vectors són unitaris, aquesta mètrica dóna un valor acotat entre 0 i 1, altrament el seu valor no està acotat.

Índex de Jaccard Considerant els vectors de característiques com a conjunts d'aquestes, aquesta mètrica es calcula amb la fórmula:

$$Jaccard(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

Aquesta mètrica dóna un valor acotat entre 0 i 1.

[1] Amb $\mathcal{P}(F)$ em refereixo al *powerset* de F, és a dir, tots els possibles subconjunts de característiques.

Índex de Jaccard ponderat Aquesta mètrica és una extensió de l'índex de Jaccard que considera una funció que assigna un pes a cada característica. El resultat dóna major importància a les característiques amb major pes:

$$WeightedJaccard(A,B) = \frac{\sum_{x \in A \cap B} weight(x)}{\sum_{y \in A \cup B} weight(y)}$$

Aquesta mètrica també dona un valor acotat entre 0 i 1.

Índex de Solapament [2] Considerant els vectors de característiques novament com a conjunts, l'índex de solapament es calcula com

$$Overlapping(A,B) = \frac{|A \cap B|}{min(|A|,|B|)}$$

Índex de Solapament ponderat De manera similar al cas de l'índex de *Jaccard*, definim l'índex de solapament ponderat de la següent manera:

$$WeightedOverlapping(A,B) = \frac{\sum_{x \in A \cap B} weight(x)}{\sum_{y \in shortest(A,B)} weight(y)}$$

El càlcul de similitud simplement correspondrà a aplicar alguna d'aquestes mètriques sobre els vectors de característiques.

3.5 Càlcul d'ordre

En els apartats anteriors s'ha treballat partint el text i el model en objectes més petits: frases i tasques. Però a part de considerar aquests de manera separada, és molt important considerar l'estructura que els uneix. En el cas dels models BPMN la seqüència de les tasques és clara i ve totalment determinada per l'estructura de graf del model. En canvi, pel text, detectar-ne l'estructura, o ordre en que succeeixen els events, és tot un problema en sí mateix. Aquest pas de l'algorisme consisteix en, donats el text i el model BPMN inicials, derivar-ne un ordre parcial dels seus elements.

Més formalment, donats dos elements d'un dels dos conjunts S i T, l'objectiu és trobar la relació $\rightsquigarrow$ que, per cada element de $S \times S$ o $T \times T$ indiqui si el primer element precedeix al segon.

3.5.1 Al text

Per poder calcular l'ordre entre els elements de S, és necessari fer una interpretació del text a nivell temporal, detectant quines frases van abans d'altres a nivell cronològic. Tot i que aquest enfoc és possible[3], queda fora de l'àmbit d'aquest projecte. Per aquest fet, hem considerat que l'ordre entre les frases del text és l'ordre d'aparició en aquest. És a dir, definim la relació d'ordre entre s_i i s_j:

$$s_i \rightsquigarrow s_j \iff i < j$$

[2] Traducció d'*Overlapping Index*

[3] Cal dir que l'ambigüitat inherent al llenguatge natural fa que els resultats obtinguts amb aquest enfoc no siguin del tot exactes.

Encara que d'entrada aquesta simplificació pugui semblar molt exagerada, cal considerar que estem tractant amb textos molt concrets. Les descripcions de procés són textos tècnics i en la seva redacció sempre és preferible la claredat. És per això que generalment es compleix aquesta propietat d'ordre.

3.5.2 Al model

En el model BPMN, per poder modelar millor aquesta relació d'ordre, adaptem la notació de graf de procés definida a [20]. Un graf de procés és una tupla $PM = (A, G, F, s, e, t)$ on:

- A és el conjunt finit d'activitats, equivalent a T
- G és el conjunt finit de *gateways*
- $N = A \cup G$
- F és el conjunt d'arestes entre els elements del model BPMN.
- $\bullet n = \{n' \in N | (n', n) \in F\}$ i $n\bullet = \{n' \in N | (n, n') \in\}$, conjunts de predecessors i successors.
- $s \in A$, l'única activitat inicial. $\bullet s = \emptyset$.
- $e \in A$, l'única activitat final. $e\bullet = \emptyset$.
- $t : G \to \{and, xor\}$, funció que associa cada gateway a un tipus concret[4].

Aquest formalisme és una simplificació de la notació BPMN, però és suficient per definir una estratègia algorítmica per determinar l'ordre entre els elements. A més, cal notar que un graf de procés i un model BPMN no són equivalents. En aquesta notació, un graf de procés és equivalent a una pool en un model BPMN (veure apartat 2.1.4).

Per completar la definició de l'ordre parcial, abans hem de definir les traces. Una traça en el model és una llista d'elements de N de la forma $s \cdot A * \cdot e$ de manera que per cada parella de nodes consecutius (n, n') en la llista tinguem que $(n, n') \in F$. Dit d'altra manera, una traça és un camí en el graf del model que va del node inicial al node final: Una possible execució del model. Definim el conjunt τ_{PM} com el conjunt de totes les possibles traces del model PM.

Així doncs, definim la relació $\preceq$ per a un procés PM com el conjunt de parells (x, y) de $(N \times N)$ tals que existeix una traça $\sigma = n_1, \cdots, n_m \in \tau_{PM}$ on $x = n_i$ i $y = n_j$ per dos naturals $i < j$ entre 1 i m. O, en altres paraules, existeix una traça on x apareix abans que i.

I finalment, partint de la definició d'ordre parcial, podem establir les tres relacions que conformen el *Behavioral Profile* d'un graf PM. Per cada parella de nodes $(x, y) \in N \times N$, x i y tenen una única d'aquestes relacions:

- Ordre estricte: $x \rightsquigarrow_{PM} y \iff x \preceq y$ i $y \npreceq x$
- Ordre estricte invers: $x \leftsquigarrow_{PM} y \iff y \preceq x$ i $x \npreceq y$

[4]Existeixen altres tipus de gateway en el formalisme del BPMN, aquests s'han de classificar en les categories *and* o *xor* en funció de si permeten l'execució simultània de múltiples branques o no.

- Exclusivitat: $x +_{PM} y \iff x \not\leq y$ i $y \not\leq x$.
- Paral·lelisme: $x||_{PM} y \iff x \leq y$ i $y \leq x$.

El behavioral profile d'un graf de procés és una informació molt important a l'hora de realitzar-ne un anàlisi, permetent trobar dependències entre tasques i determinar quines accions es poden dur a terme en paral·lel.

Veiem doncs com la relació que buscàvem és un subconjunt del *Behavioral Profile* del graf de procés del BPMN en qüestió.

Tot i això, aquest enfoc només serveix per establir l'ordre entre els elements d'un graf de procés. Aquest projecte pretén analitzar qualsevol model BPMN, i els models amb múltiples pools defineixen múltiples grafs de procés per a un sol model, que es comuniquen a partir de message flows (veure apartat 2.1.4). Com que aquest enfoc només té en compte els sequence flows –Arcs en el graf de procés– cal un post-processat addicional per tenir en compte la informació que ens proporcionen els missatges.

Suposem que tenim un model BPMN amb diferents pools, o grafs de procés, $BPMN = PM_1, \cdots, PM_K$ i un conjunt de message flows $MF = \{(n_1, n_2)|n1 \in PM_i \wedge n_2 \in PM_j \wedge i \neq j\}$[5]. A més, definim per a tot node n els conjunts $After_n = \{n' \in N|n \rightsquigarrow n'\}$ i $After_n = \{n' \in N|n' \rightsquigarrow n\}$. El post-processat correspon al següent algorisme:

1. Creem un nou behavioral profile BF amb els nodes de $PM_1 \cdots PM_K$, i considerem totes les relacions entre elements de diferents grafs de procés com a $+$.
2. Per cada message flow $(n_i, n_j) \in MF$:
   (a) Modifiquem a BF la relació entre n_i i n_a per $\rightsquigarrow$ per tot $n_a \in After_{n_j} \cup \{nj\}$
   (b) Modifiquem a BF la relació entre n_j i n_b per $\leftsquigarrow$ per tot $n_b \in Before_{n_i} \cup \{ni\}$
   (c) Afegim també les relacions simètriques, si escau.
3. Repetir el punt anterior fins que el behavioral profile convergeixi (no es modifiquen relacions noves)

Aquest algorisme iteratiu permet crear un *behavioral profile* general per a tot el model BPMN. Després del post-processat, tenim un ordre definit entre alguns dels elements del BPMN. Això és així perquè no sempre és possible establir un ordre entre dos elements d'un model. Els elements pels quals no s'ha pogut establir l'ordre tindran la relació d'exclusivitat en acabar l'algorisme.

3.6 Càlcul de la correspondència òptima

El pas final és obtenir la correspondència òptima entre frases del text i tasques del model BPMN. De forma similar a [1], definim la correspondència òptima com la funció $f_{CO} : T \to S$ que compleix:

- Assignació total: El domini de f_{CO} és T, és a dir, totes les tasques tenen imatge.
- Optimalitat: Es maximitza la suma $\sum_{t \in T} S(f_{Task}(t), f_{Sentence}(f_{CO}(t)))$, on S és una de les funcions de similitud de l'apartat 3.4.

[5]En aquesta expressió s'utilitza $n \in PM$ per dir que n pertany al conjunt de nodes de PM.

- Consistència d'ordre: $s = f_{CO}(t) \wedge s' = f_{CO}(t') \wedge t \preceq t' \implies s \preceq s'$.

D'entrada el fet d'assignar frases a tasques, i no a l'inrevés, pot semblar arbitrari. No obstant això, la idea que hi ha darrere té a veure amb l'estructura implícita dels models BPMN i els textos. Les tasques d'un model BPMN estan pensades per definir una única acció atòmica. D'altra banda, una frase en un text pot contenir la descripció de diverses accions, ja sigui mitjançant oracions coordinades o subordinades, o fins i tot amb informació implícita. És per aquest fet que s'ha escollit fer l'assignació de frases a tasques, ja que el cas més habitual serà que una frase pugui referir-se a més d'una tasca, mentre que el cas contrari és menys probable i seria en tot cas un error. Una tercera opció seria considerar, ja no una funció, sinó una llista de parelles frase-tasca. Així es contemplaria el cas d'una tasca referint-se a dues frases i viceversa. Aquest tercer enfoc, però, té l'inconvenient que la restricció d'optimització preferirà el màxim nombre de parelles possibles, assignant totes les tasques a totes les frases.

És també important considerar si aquest conjunt de restriccions pot donar lloc a un problema irresoluble. Vegem que no es pot donar mai aquest cas:

Demostració. Sigui $S = \{s_1, \cdots, s_N\}$, $T = \{t_1, \cdots, t_M\}$ una instància qualsevol del problema. Considerem l'assignació f_C tal que $f_C(t) = s_N$, $\forall t \in T$. Aquesta assignació f_C compleix les restriccions d'assignació total i consistència d'ordre, ja que d'una banda totes les tasques tenen imatge i per altra banda $\forall t, t' : f_C(t) \preceq f_C(t')$, ja que totes les tasques s'assignen a la mateixa frase. Si f_C no compleix la restricció d'optimalitat, vol dir que existeix una $f'_C = f_{CO} \neq f_C$, altrament $f_C = f_{CO}$. Veiem, doncs, com sempre existeix una solució al problema. □

Capítol 4

Implementació

4.1 Estructura general de l'algorisme

L'algorisme per comparar descripcions textuals i models BPMN s'ha implementat seguint una estructura similar a la formalització proposada a l'apartat 3. A la figura 4.1 es pot veure l'estructura de mòduls que conformen l'algorisme.

Per consultar el codi font del programa, veure l'annex B.

4.2 Pre-processat de les dades d'entrada

El primer pas que realitza l'algorisme és un pre-tractament de les dades d'entrada perquè es puguin analitzar fàcilment a les següents fases d'aquest. En aquesta secció es descriu el pre-processat que segueixen les dades d'entrada de l'algorisme: El text pla en llenguatge natural i el model BPMN.

4.2.1 El text analitzat

Pel que fa a l'anàlisi del text, l'objectiu és obtenir una estructura de dades que reflecteixi l'estructura sintàctica d'aquest a partir del text pla d'entrada.

Implementar aquesta tasca no és un dels objectius d'aquest projecte, sinó que s'usa la llibreria *Freeling* per al processat del text en llenguatge natural, deixant per a aquest projecte la tasca d'interpretar aquesta informació. En comptes d'integrar *Freeling* com una llibreria en el projecte, s'utilitza la API http del *Textserver*. La decisió d'utilitzar el *Textserver* en comptes de *Freeling* com una llibreria nativa ve motivada per tres raons. La primera és el pes de la llibreria, ja que requereix els diccionaris de Wordnet en diversos idiomes, entre altres coses, i el seu pes compilat acaba sent de diversos GB. Aquest problema es veuria accentuat pel fet que *Freeling* està programat en C++, cosa que implicaria haver d'incloure'l com diversos binaris compilats per a diversos sistemes operatius i arquitectures si es vol una solució portable en Java, multiplicant encara més el pes del programa final. La segona raó és la simplicitat de programació. Tot i que *Freeling* ofereix una interfície nativa per a java (JNI), resulta més senzill navegar una única estructura de dades en un format estàndard com JSON. Finalment, la tercera raó

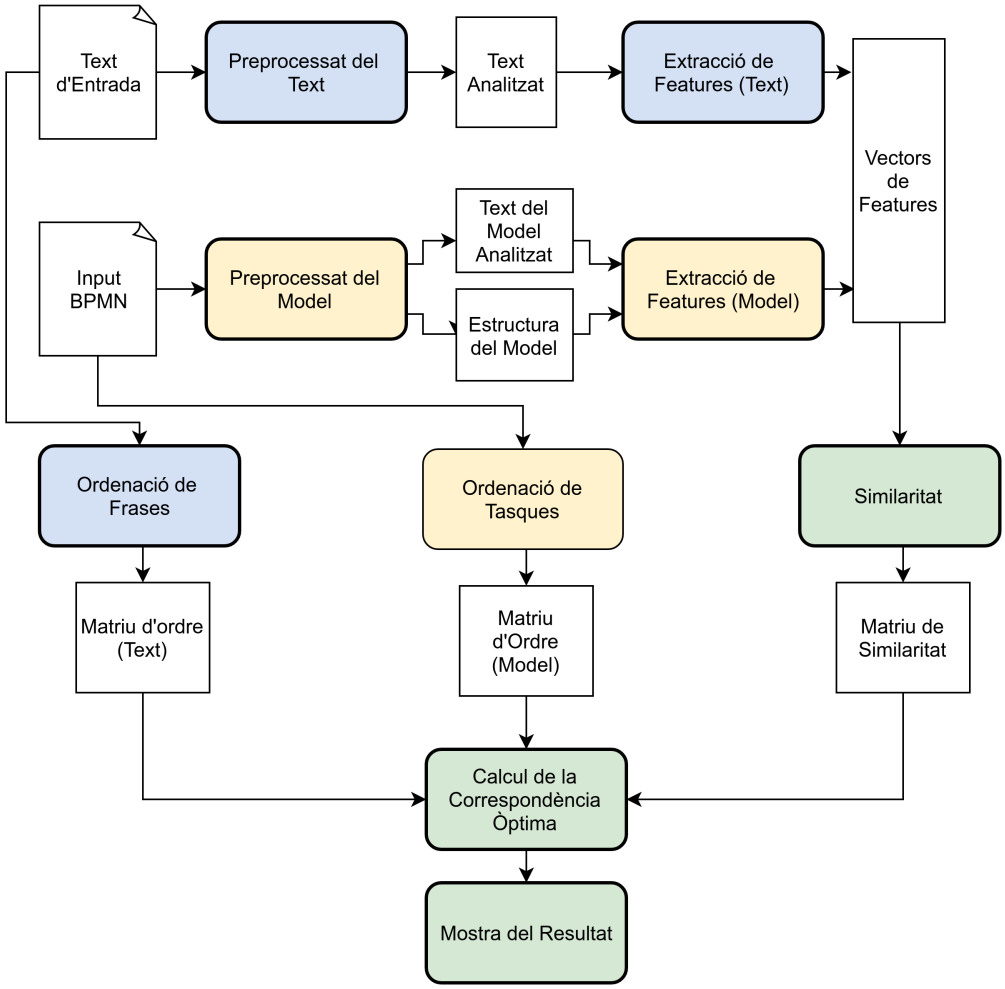


Figura 4.1: Estructura general de la implementació de l'algorisme.

és un tema de conveni. La resta de projectes de NLP4BPMN ja utilitzaven el *Textserver* abans de començar aquest projecte.

Per l'anàlisi del text es crida al servei *semgraph* del *Textserver* que realitza l'anàlisi més complet del text disponible. Aquest anàlisi conté, entre altres coses:

- El text separat en frases i l'anàlisi morfològic de cadascuna de les paraules.
- La llista de predicats[1] que conté cadascuna de les frases.
- L'arbre de constituents de les frases.
- L'arbre de dependències de les frases.
- El graf semàntic del text.

[1]No només els predicats verbals, sinó totes aquelles parts de la frase que impliquin alguna acció, es consideren predicats en aquest cas.

4.2.2 El model BPMN

Pel que fa al model BPMN, s'han considerat dues parts diferenciades. Per una banda, el BPMN és principalment informació estructurada: El graf de processos, els diferents elements i les relacions jeràrquiques entre tasques, pools i swimlanes. D'altra banda, la majoria dels elements del BPMN contenen text en llenguatge natural que ofereixen la informació semàntica. És per això que s'han plantejat dos pre-processats diferents per al BPMN.

El primer pre-processat consisteix simplement a obtenir la informació estructural del BPMN i representar-la en un format adient. Per a fer la lectura del model, s'usa el parser de la llibreria Activiti, que ja implementa la lectura dels models BPMN. No obstant això, Activiti no és una llibreria pensada per llegir BPMNs, sinó per executar-los. És per això que les funcionalitats que exposa estan enfocades a un altre tipus de programes. Això ha fet que calgués un posterior refinament de les estructures de dades, adaptant-les a quelcom més adient per al problema. L'alternativa a Activiti hagués estat implementar la lectura dels models BPMN, tot i això, s'ha desestimat per evitar els possibles contratemps que els errors en un parser propi poguessin introduir.

El segon pre-processat correspon a analitzar el text, en llenguatge natural, contingut al model. Per a fer-ho, s'usa novament la llibreria *Freeling*. Tot i això, alguns dels processats que realitza la llibreria, com la detecció de *senses* requereixen d'un cert context. Per exemple, en català, la paraula "gat" se sol referir típicament a l'animal, però si en el text apareixen paraules com "taller", "cotxe" i "reparació", *Freeling* farà servir aquesta informació de context per determinar que no es tracta pas de l'animal, sinó de l'eina. Com que les frases del model BPMN típicament són senzilles, i per tant no contenen prou context, s'hi han afegit les frases del text a l'anàlisi. Així, si el vocabulari del text i el model estan en un àmbit similar, la detecció del significat de cada paraula es realitzarà correctament. Finalment, les frases del text original es descarten i queda el text del model analitzat.

4.3 Extracció de característiques

Un cop la informació d'entrada ha estat degudament pre-processada, el següent pas que realitza l'algorisme és l'extracció de característiques. Una vegada més, la tasca s'ha dividit en dos mòduls separats per al text i el model, ja que la seva representació encara no és uniforme.

En el mòdul d'extracció de característiques es declaren quines són les característiques que s'extrauran del text i del model. Cada característica es defineix amb un pes, això permet donar més importància a que el text i el model comparteixin subjecte que no pas que comparteixin una paraula. Aquest pes s'utilitza després en el càlcul de similitud i està determinat manualment.

Aquest és el conjunt de característiques que s'han considerat a l'hora d'implementar l'algorisme[2]. Seguint el formalisme descrit a l'apartat 3.3, es descriu cada tipus de característica amb el nom i arguments.

Has-Lemma(pos, lemma): El text conté una paraula amb arrel **lemma** i categoria gramatical **pos**.

[2]Els noms que es fan servir són els mateixos que en el codi, per facilitar la comprensió d'aquest.

Has-action(a): El text descriu una acció representada per la paraula **a**, generalment un verb.

Has-Synset(s): El text conté una paraula que pertany al synset **s** al diccionari de Wordnet 3.0[3]

Has-Parent-Synset(s): El text una paraula el synset de la qual és un homònim del synset **s**. És a dir, la paraula i **s** tenen una relació de *és un*.

In-Agent(lemma, pos, verb): El rol d'agent de l'acció indicada per **verb** conté la paraula amb arrel **lemma** de categoria gramatical **pos**

In-Patient(lemma, pos, verb): El rol de pacient de l'acció indicada per **verb** conté la paraula amb arrel **lemma** de categoria gramatical **pos**

Lemma-conditional-pred(lemma, pos) L'element es troba a continuació d'una clausula condicional[4] que conté la paraula amb arrel **lemma** de categoria gramatical **pos**.

El més important d'aquestes característiques és que han de ser rellevants tant pel text com pel model. Es podria extreure informació molt més complexa de cadascun dels dos per separat, però no existiria una versió homòloga a l'altre.

És important notar també que moltes vegades s'inclou informació addicional com la categoria gramatical o el verb, en el cas d'agents i pacients. Això es fa per concretar les característiques. Per exemple, si dos fragments de text contenen la paraula "document" –en anglès– però en una apareix com a verb i l'altre com a nom, no s'hauria de comptar com la mateixa característica.

La característica **Has-Parent-Synset** s'utilitza per detectar paraules que no són sinònims directes. Per exemple, es pot donar el cas que el text parli d'un "document" que s'ha dit que és una carta, però al model s'utilitzi sempre la paraula "carta". Quan el model parli de carta i el text de document, ni les paraules ni els synsets coincidiran[5] però si que ho farà algun dels seus hiperònims. No obstant això, per evitar crear falsos positius amb aquestes característiques, cada cop que es va amunt en la cadena d'hiperonímia, el pes de la característica es multiplica per un cert valor fent que vagi disminuint. Aquest multiplicador és un dels paràmetres configurables de l'algorisme: "Hiperonimy multiplier".

La característica **In-Agent** serveix per identificar el rols semàntics d'agent en una frase o tasca. L'agent d'una acció respon a la pregunta "Qui?", és a dir, indica qui realitza l'acció. L'extracció d'aquesta característica és molt diferent entre el text i el model. En una frase, identificar l'agent és una tasca purament de NLP i s'ha implementat utilitzant Freeling. D'altra banda, per identificar l'agent d'una tasca en el model, es fa servir la semàntica del BPMN. En un model BPMN l'agent que realitza cadascuna de les accions s'ha d'explicitar mitjançant les *pools* i les *swimlanes*, així que la informació s'obté directament del model.

Finalment, per cadascuna de les característiques declarades, s'implementa una funció extractora pel text i pel model. És a dir, una funció que, rebent el text analitzat en

[3]La versió del diccionari pot canviar, però és la que utilitza Freeling i per tant la que s'utilitza en aquest projecte.

[4]Ja sigui indicada amb llenguatge natural: "if ... then ...", o bé utilitzant una gateway exclusiva, pel cas del model.

[5]Carta i document no són sinònims en general, tot i que puguin ser-ho en el text. Per tant, no comparteixen el mateix synset.

retorna un vector de característiques i anàlogament pel model. És important notar que el text i el model són estructures totalment diferents, i que l'objectiu d'aquest mòdul és crear una representació comú. Així doncs, és necessari tenir per duplicat cadascuna de les funcions extractores, pel text i pel model, ja que el codi de cadascuna d'aquestes és conceptualment molt diferent. El resultat d'aquestes funcions extractores correspon al resultat del mòdul d'extracció.

4.4 Càlcul de la similitud

Establir una noció de similitud entre frases del text i taques del model és complicat. No obstant, un cop reduït el problema a vectors de característiques, existeixen tot tipus de mètriques a la literatura. En concret, la implementació d'aquest mòdul ha consistit a implementar les diferents mètriques de similitud descrites a l'apartat 3.4. la implementació és una traducció directa de les fórmules.

Aquest mòdul també permet escollir, mitjançant una opció de configuració, quina és la mètrica de similitud que es vol exposar.

Com a valor per defecte s'ha escollit utilitzar l'índex d'Overlapping ponderat. Degut a la naturalesa del problema amb el que es tracta, les frases del text sempre solen ser més verboses que les frases del model i, per tant, contenen més paraules. Això fa que el conjunt de característiques de les frases del text sigui quasi sempre més gran que el de les frases del model. En les altres funcions de similitud, aquesta diferència de cardinalitat fa que el valor, tot i ser correcte a nivell relatiu, acabi tenint un valor molt baix en magnitud. L'índex d'Overlapping té la propietat que dos conjunts tenen una similitud màxima de 1 si el petit és subconjunt del gran. Per aquest fet, l'índex d'Overlapping, i la seva versió ponderada, donen resultats numèrics molt més intuïtius[6] de cara a mostrar una puntuació a l'usuari final. No obstant això, l'objectiu no és sempre donar una puntuació intuïtiva. És possible que a l'hora de comparar, els resultats obtinguts d'altres mètriques de similituds siguin més precisos a nivell relatiu. Per això s'ha escollit deixar la funció com a paràmetre configurable.

El resultat d'aquest mòdul és, finalment, la matriu de similitud. La matriu de similitud és una matriu $|S| \times |T|$ l'element (i, j) de la qual indica la similitud entre la frase i i la tasca j.

4.5 Ordenació de models i textos

El mòdul d'ordenació s'estructura, novament, en dues parts. L'ordenació de textos i de models. L'ordenació de textos, tal com s'explica a l'apartat 3.5.1, és una simplificació respecte el problema real i la implementació ha resultat molt senzilla.

El resultat del mòdul son dues matrius, una pel text i una pel model. En aquestes dues matrius, cada element pot prendre els valors: $\rightsquigarrow$, $\leftsquigarrow$ i $\neq$. Els dos primers valors representen les relacions "Precedeix a" i "Succeeix a". Així doncs, la matriu indica per cada cada element (i, j) la relació d'ordre entre l'element i i el j.

[6] Utilitzant cosinus o jaccard ponderat, la similitud d'una parella de model-text bona és de l'ordre de 0.1, mentre que si la parella és dolenta el valor de l'ordre de 0.001. Utilitzant l'índex d'Overlapping aquests valors passen a ser 0.5 i 0.01.

4.5.1 Al model

Com s'explica a la secció 3.5.2 l'ordenació de tasques al model es fa mitjançant el càlcul del *Behavioral Profile*[20] del model. Cal notar que si s'intentés implementar el càlcul a partir de la definició formal, el cost d'aquest algorisme seria exponencial, havent de tractar tots els possibles camins del graf de procés. No obstant, existeixen algorismes molt més eficients a la literatura. En aquest projecte s'ha usat la llibreria JBPT (Business Process Technologies for Java) que implementa un algorisme eficient pel càlcul dels behavioral profiles. També s'ha implementat l'algorisme de post-processat, que s'ha implementat tal com es defineix a la secció 3.5.2.

Tot i això, els *Behavioral Profiles* no van ser la primera alternativa estudiada. Primer de tot, es va plantejar aquest problema com una ordenació topològica del graf de processos. Com que els grafs de processos no son acíclics, per a plantejar aquest enfoc és necessari un algorisme d'eliminació de *back-edges* considerant el BPMN un graf de flux[4]. Aquest procediment està implementat al projecte, però es va descartar ja que els resultats que donava no eren exactament el tipus d'ordre que es volia modelar.

4.6 Càlcul de la correspondència òptima

L'objectiu d'aquest mòdul és calcular una aplicació de tasques a frases que satisfaci les tres restriccions definides a l'apartat 3.6: Assignació total, Optimalitat i Consistència d'Ordre. La primera és una propietat inherent al problema. La segona fa referència a la matriu de similituds. La tercera es pot veure com un conjunt de restriccions en funció de la matriu d'ordre.

Per implementar aquest apartat s'han plantejat diverses alternatives. L'opció escollida finalment ha estat el modelar el problema com un programa ILP (Integer Linear Programming). No obstant això les tres solucions estudiades tenen els seus punts forts i febles. Com es pot veure a la figura 4.2, LAP destaca per ser el més eficient dels tres algorismes, tot i trobar solucions de menor qualitat. D'altra banda, la qualitat de les solucions amb CP i ILP és la mateixa. No obstant això, donada la major flexibilitat que ofereix Constraint Programming, aquest es planteja com a solució alternativa en cas que noves restriccions introduïdes al problema en un futur fessin que ILP deixés de ser una alternativa viable.

En els següents apartats es descriu en més detall cadascuna de les diferents alternatives plantejades i es justifica l'elecció de ILP com a millor opció per a resoldre el problema.

4.6.1 Reducció a LAP

El LAP (Linear Assignment Problem)[16] es pot formular, informalment, de la següent manera:

> Una instància del problema correspon a un conjunt A de N d'*agents* i un conjunt de T de N *tasques*. Per cada agent es disposa del cost que suposa assignar-lo a una tasca concreta. L'objectiu és trobar l'assignació que assigna cada agent a una tasca diferent minimitzant el cost total.

Com es pot veure, aquest problema guarda una certa relació amb el que s'ha de resoldre en aquest cas. Tots dos problemes busquen trobar una assignació entre elements

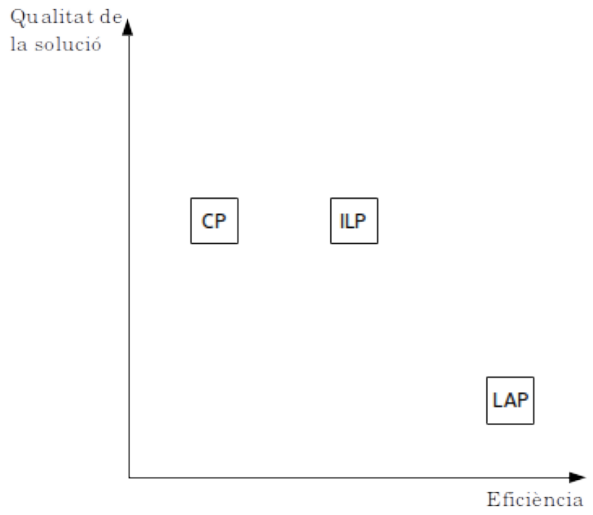


Figura 4.2: Comparativa dels diferents algorismes estudiats.

de dos conjunts tot minimitzant una funció objectiu. Tot i això, un LAP no contempla cap mena de consistència d'ordre.

La solució proposada és adaptar les condicions del problema original per aconseguir una instància d'un LAP. Per fer-ho, calen dues coses: Codificar l'ordre i aconseguir el mateix nombre de tasques i frases. Per codificar l'ordre, cal canviar el concepte d'ordre pel de distància, multiplicant la similitud entre cada tasca i frase, per la inversa de la distància entre aquests. La intuïció darrere ponderar per la distància inversa és penalitzar solucions on, per exemple, s'assignen dues frases consecutives a dues tasques molt separades en el model; creant una noció d'ordre similar a la de la restricció original. Pel que fa a tenir el mateix nombre de tasques i frases, es pot resoldre ampliant el conjunt més petit amb elements falsos amb similitud 0 per qualsevol altre element. Així, l'algorisme assignarà els pitjors candidats del matching a aquests elements falsos.

La motivació darrere aquesta idea és que LAP té solució en temps polinòmic, cosa que faria que tot l'algorisme es pogués executar en temps polinòmic. Si la relaxació de les condicions resulta ser massa forta, el resultat del LAP es podria seguir utilitzant com a solució inicial per algun algorisme d'optimització global.

Tot i això, un cop portada a la pràctica, aquesta solució no va donar els resultats esperats. El problema és que el resultat del LAP ha de ser una assignació bijectiva. Mentre que la naturalesa d'aquest problema fa que, moltes vegades, l'assignació òptima sigui justament no bijectiva. Per exemple, poden existir frases en el text que serveixin d'introducció per al lector i que no es refereixin a cap acció concreta. D'altra banda, també és comú el cas que una frase correspongui a dues tasques (una conjunció de dues accions, per exemple). Aquestes diferències entre la naturalesa dels problemes han fet que es descartés aquesta opció. Tot i això, si la mida del problema fos suficientment gran, es pot optar per utilitzar LAP intercanviant eficiència per optimalitat.

4.6.2 Constraint Programming

La segona alternativa plantejada consisteix a modelar el problema com un conjunt de restriccions per a un solver de Constraint Programming[8]. Una bona alternativa per

a aquesta tasca és Choco Solver, una llibreria de software lliure per a Java que permet modelar problemes de satisfacció de restriccions.

Per modelar el problema com un CSP s'han estudiat dues alternatives.

La primera alternativa defineix el conjunt de variables T com el conjunt de tasques, cadascuna amb el domini de valors $1 \cdots |S|$, on $|S|$ és el nombre total de tasques. Es considera que la frase i i la tasca j estan relacionades si $T_j = i$. A més, s'afegeixen les variables auxiliars $Sim_1 \cdots Sim_{|T|}$, on $|T|$ és el nombre de frases. A les variables se'ls hi imposen les restriccions següents:

- Per cada tasca T_i: $T_i = j \implies Sim_i = SM_{ij}$, on SM és la matriu de similituds.
- Per cada parella de tasques T_i, T_j una restricció: $TO_{ij} = \rightsquigarrow \implies T_i <= T_j$, on TO és la matriu d'ordre del model.
- A més, s'indica al solver que maximitzi l'expressió: $\sum_i^{|S|} Sim_i$.

És fàcil veure que aquesta codificació compleix amb totes les restriccions imposades. La optimalitat la garantitza el solver maximitzant la suma de les similituds, mentre que la consistència d'ordre la garanteix la segona restricció. D'aquesta segona restricció, cal notar que es fa servir el truc que les frases tenen un ordre seqüencial per considerar l'ordre entre frases igual a l'ordre dels enters que les representen.

La segona alternativa fa servir variables booleanes A_{ij} = "La tasca i s'ha assignat a la frase j". En aquesta nova codificació, les restriccions són:

- Per cada valor i: $\sum_j A_{ij} = 1$.
- Per cada parella (s, s') de frases i cada parella (t, t') de tasques: $(TO_{tt'} = \rightsquigarrow$ i $SO_{ss'} = \leftsquigarrow \implies \neg(A_{s,t} \wedge A_{s',t'})$, on TO i SO son les matrius d'ordre del model i el text respectivament.
- A més, s'indica al solver maximitzar l'expressió $\sum_i^{|S|} \sum_j^{|T|} A_{ij} * SM_{ij}$

Les dues alternatives donen un resultat equivalent, i compleixen amb les restriccions del problema. No obstant això, el temps d'execució d'algunes instàncies del problema amb Choco Solver són massa elevats, arribant a trigar fins a 7 minuts. La diferència de temps entre les dues alternatives no és significativa. Aquest temps d'execució és massa gran pel tipus de software que es vol desenvolupar, per això es va descartar Constraint Programming com alternativa en general. Tot i això, si les restriccions del problema fossin més complexes seria interessant tornar a estudiar aquesta alternativa, ja que Constraint Programming és un enfoc totalment genèric a resoldre problemes de restriccions que permet una màxima flexibilitat i és molt senzill d'utilitzar.

4.6.3 Integer Linear Programming

Un subconjunt dels problemes de satisfacció de restriccions és la programació entera. Un programa per a un solver de ILP[19], o *linear program*, és un conjunt d'inequacions lineals definides sobre un conjunt de variables enteres[7].

Partint de la segona formulació plantejada en el mètode de Constraint Programming, un conjunt de variables booleanes A_{ij}, es poden redefinir les restriccions del problema

[7] O booleanes, com a cas especial dels enters amb domini $\{0, 1\}$

anterior com un conjunt d'equacions[8] i inequacions lineals. Fixant-nos en les tres restriccions de l'apartat anterior, la 1 i la 3 ja són equacions enteres, així doncs no cal fer res. Pel que fa a la segona restricció, l'expressió booleana $\neg(A_{s,t} \wedge A_{s',t'})$ es pot reinterpretar com: $A_{s,t} + A_{s',t'} \leq 1$. Aquesta transformació permet adaptar la formulació de l'apartat anterior a un *linear program*.

Aquesta codificació del problema és equivalent a les de l'apartat anterior, i per tant compleix les restriccions imposades. D'altra banda utilitzant un solver senzill de ILP (*lp_solve*), els temps d'execució pels problemes difícils es redueixen de minuts a fraccions de segon, complint perfectament les expectatives del software que es vol obtenir. Aquesta reducció tant dràstica de temps es dóna ja que el solver de ILP és capaç d'assumir més coses sobre l'espai de cerca, ja que el problema es planteja de manera molt més concreta que no pas amb Constraint Programming.

4.7 Mostra dels resultats

La funció d'aquest últim mòdul és recopilar tota la informació obtinguda per a mostrar-la a l'usuari. Com més informació es mostri del funcionament intern de l'algorisme, més senzill serà per a l'usuari identificar els problemes entre el text i el model i com augmentar la similitud. Per això, la sortida de l'algorisme correspon a una estructura amb els següents elements:

- La puntuació de similitud general entre el model i el BPMN [9], valor entre 0 i 1.
- La assignació: Per cada tasca, la frase a la que aquesta correspon i la puntuació d'aquesta.
- Per cada assignació, les característiques comunes que han coincidit a la tasca i la frase, i per tant han augmentat la similitud.

A més, es descartaran de l'assignació aquelles parelles frase-tasca que tinguin un valor de similitud menor que un cert llindar. Aquestes tasques s'assignaran a una secció "Discarded", que no correspon a cap frase. El llindar es determina actualment com $mean - k \cdot mad$ on $mean$ és la mitjana de les puntuacions, mad és la desviació mitjana de la mostra de les puntuacions, i k és un paràmetre configurable de l'algorisme amb el nom "Threshold multiplier".

Per mostrar les característiques s'utilitzen les frases explicatives declarades juntament amb la característica. Així, si tant a la frase com a la tasca apareix la característica **Has-Lemma(secretary, noun)**, l'usuari veurà un text més intuïtiu: "Contains the noun *secretary*". Aquestes frases explicatives es poden canviar al fitxer de configuració del programa.

4.8 Llenguatge de programació

El llenguatge de programació és un punt important a considerar a l'hora d'implementar qualsevol software. Diferents tipus de llenguatges ofereixen un nivell diferent d'abstrac-

[8] Afegint variables auxiliars addicionals, els solvers de ILP també permeten modelar equacions, aquest pas el realitza automàticament el solver.

[9] Calculada com la suma de les similituds de cada assignació de tasca a frase dividida per el nombre total de tasques.

ció i expressivitat. D'altra banda, el llenguatge també pot determinar en gran mesura el rendiment d'un codi. Un altre aspecte a considerar és la compatibilitat i portabilitat que requereix el producte esperat. A l'hora de decidir el llenguatge de programació per aquest projecte, s'han tingut en compte aquests factors. En aquesta secció s'ofereix una justificació del fet d'haver escollit *Clojure* com a llenguatge de programació principal del projecte.

Clojure és un llenguatge de programació funcional i dinàmic[10] per a la JVM[11]. Això fa que els programes en Clojure siguin totalment compatibles amb programes fets en Java en tots dos sentits. D'altra banda, Clojure és també un dialecte modern de LISP. Això per una banda mostra els típics avantatges que aquesta família de llenguatges ofereix: Macros, homoiconicitat, funcions *first-class*, etc. D'altra banda, Clojure es diferencia d'altres dialectes de LISP oferint una varietat major d'estructures de dades: No tot són llistes. A més, amb la característica que totes les seves estructures de dades són immutables[12].

L'expressivitat és el punt fort de Clojure. En aquest projecte es tracta amb estructures de dades dinàmiques i complexes. Per exemple: Ha estat típic en el codi el recórrer diverses vegades l'estructura JSON que retorna el *Textserver* buscant diferents patrons, a l'hora d'extreure característiques. Com que aquest llenguatge permet tractar tot tipus de formats típics (JSON, XML, etc) com estructures de dades natives del llenguatge, s'ha pogut treballar directament sobre aquestes estructures sense haver d'introduir una nova capa de compatibilitat. De fet, la flexibilitat del llenguatge i la llibreria *Spectre* han permès estructurar bona part del codi d'extracció de característiques en forma de queries declaratives sobre l'estructura JSON, com si es tractés de les operacions SELECT i UPDATE típiques de SQL, utilitzant exactament la mateixa sintaxi que la del llenguatge base.

Pel que fa al rendiment, Clojure és aproximadament tant ràpid com Java si es limita l'ús de certes funcionalitats. Quan s'utilitzen construccions de molt alt nivell, aquestes poden portar un cert *overhead*. En tot cas, a nivell de temps d'execució, podem dir que Clojure està al nivell de Java dins un ordre de magnitud. D'altra banda, el que sí que resulta un problema amb Clojure és el temps d'inicialització[13]. Per arrancar un programa cal carregar primer el runtime de Java i després del runtime de Clojure, cosa que porta uns segons. Això fa que aquest llenguatge no sigui adient com a llenguatge de scripting. No obstant això, aquest no és el cas d'aquest projecte. El programa es carregarà una vegada en un servidor i no pararà d'executar-se, servint diverses peticions a mode de servei web. En un cas així, aquest temps d'inicialització es pot considerar negligible.

Finalment, un dels punts decisius a escollir Clojure ha estat la compatibilitat. Aquest projecte forma part d'un projecte més gran, on la major part del codi està fet en Java, i els diferents algorismes s'han integrat en un servidor web també en Java. Donades les necessitats de compatibilitat amb Java, s'ha escollit un llenguatge totalment compatible amb aquest.

[10] Tot i suportar tipat estàtic amb un sistema de tipus similar al de Haskell, aquest és opcional i no es fa servir per aquest projecte.

[11] Java Virtual Machine, l'intèrpret de bytecode que permet executar programes en Java bytecode. Llenguatges com Java, Clojure o Scala compilen per al set d'instruccions de la JVM.

[12] Una estructura de dades immutable és aquella que no pot ser modificada una vegada s'ha creat. En comptes de modificar-la se'n crea una de nova.

[13] Aquest problema ja és present a Java, però amb Clojure empitjora.

Capítol 5

Avaluació dels resultats

Un cop desenvolupat l'algorisme cal una manera de validar els resultats obtinguts. En aquest apartat es plantegen diversos experiments, cadascun dissenyat per avaluar una característica concreta de l'algorisme.

Durant els experiments, es farà referència a les parelles model-text de l'annex A utilitzant el codi de quatre caràcters indicat en cadascuna. Així, per exemple, la parella "Computer Repair Shop" s'abreviarà utilitzant `[COMP]`. A més, es referirà a traces detallades de l'execució de l'algorisme de l'annex C quan resulti necessari.

5.1 Aparellament de models i textos

5.1.1 Plantejament

L'objectiu d'aquest experiment és utilitzar l'algorisme per aparellar models i descripcions textuals. Es pretén observar tant tant la qualitat com l'eficiència[1] de l'algorisme a l'hora de comparar textos i models que no parlin del mateix procés. La motivació d'aquest experiment és observar com es comporta l'algorisme davant d'exemples de model i text que clarament no parlen del mateix, per veure si l'algorisme seria capaç de descartar-los. S'espera que, donats un conjunt de textos en llenguatge natural i un conjunt de models BPMN, l'algorisme sigui capaç de determinar quin text correspon a cada model.

5.1.2 Metodologia

Es disposa d'un benchmark amb un conjunt T de textos en llenguatge natural i un conjunt M de models BPMN. En total hi ha 9 models i textos aparellats, és a dir, per cadascun dels textos de T, hi ha un model de M que parla del mateix que aquest. Tenint aquest fet en compte, considerem que existeix una assignació òptima d'elements de T a M de manera que cada text correspon al model que parla del mateix.

Les 9 parelles model-text utilitzades son les que hi ha disponibles a l'annex A, tret del cas de `[ZOO]` que es reserva per l'experiment de l'apartat 5.3.

[1] Existeix la possibilitat que les instàncies del problema siguin més difícils si els textos i els models que s'intenten comparar no parlen del mateix. Aquest experiment pretén mesurar això a més de l'objectiu esmentat.

Donat aquest escenari, l'experiment es durà a terme de la següent manera:

1. Calcular, per cada parella (t, m) amb $t \in T$ i $m \in M$, la puntuació de similitud utilitzant l'algorisme.
2. Mesurar els temps de cadascuna de les execucions de l'algorisme.
3. Determinar l'assignació $A : T \to M$ de manera que cada element de T s'assigna a l'element de M pel qual $sim(t, m)$ és màxima.

L'objectiu és determinar si l'assignació òptima i A són la mateixa assignació. A més, l'experiment també pretén veure si existeix una correlació entre la similitud dels models i el temps d'execució, per això s'utilitzaran els temps mesurats.

5.1.3 Resultats

La taula 5.1 mostra per cada parella de text (files) i model (columnes) quina és la similitud global entre els dos models. S'usa la mètrica de similitud per defecte: Weighted Overlapping Index. La taula 5.2 mostra els temps d'execució de calcular l'assignació òptima per cadascuna de les parelles.

T \M	[BICL]	[COMP]	[CRED]	[DISP]	[HOSP]	[HOTL]	[RECO]	[REST]	[UNWR]
[BICL]	**0.346**	0.197	0.002	0.025	0.038	0.088	0.178	0.025	0.229
[COMP]	0.042	**0.285**	0.005	0.069	0.037	0.071	0.038	0.035	0.236
[CRED]	0.035	0.050	0.114	0.042	0.050	0.060	**0.184**	0.024	0.165
[DISP]	0.033	0.046	0.011	**0.315**	0.054	0.066	0.194	0.021	0.194
[HOSP]	0.040	0.087	0.089	0.042	**0.491**	0.101	0.183	0.064	0.168
[HOTL]	0.030	0.048	0.003	0.018	0.039	**0.351**	0.007	0.039	0.180
[RECO]	0.016	0.028	0.085	0.012	0.039	0.032	**0.328**	0.033	0.146
[REST]	0.040	0.027	0.006	0.021	0.035	0.146	0.010	**0.480**	0.164
[UNWR]	0.026	0.049	0.004	0.024	0.042	0.084	0.029	0.017	**0.339**

Taula 5.1: similitud per les diferents parelles de text (fila) i model (columna)

T \M	[BICL]	[COMP]	[CRED]	[DISP]	[HOSP]	[HOTL]	[RECO]	[REST]	[UNWR]
[BICL]	290	232	278	242	388	468	312	942	261
[COMP]	210	167	188	216	285	298	236	441	212
[CRED]	274	229	287	288	405	487	302	766	239
[DISP]	199	141	169	211	247	275	183	483	182
[HOSP]	385	355	425	402	579	671	507	1398	383
[HOTL]	261	220	267	237	362	473	299	800	283
[RECO]	195	170	206	161	316	343	229	582	219
[REST]	283	268	333	277	388	542	343	1102	300
[UNWR]	326	278	337	332	452	535	368	1034	438

Taula 5.2: Temps d'execució per computar la similitud per cada parella de text (fila) i model (columna) en mil·lisegons.

A més, per cadascuna de les files de la taula de similituds, s'ha calculat el ràtio entre l'element màxim i la mitjana dels de més l'objectiu de quantificar el nivell de confiança

Text	$max/avg(row - max)$
[BICL]	3.980
[COMP]	4.808
[CRED]	3.052
[DISP]	4.581
[HOSP]	5.697
[HOTL]	8.708
[RECO]	7.548
[REST]	9.630
[UNWR]	11.065

Taula 5.3: Nivell de confiança de l'assignació de les diferents parelles model-text

amb que l'algorisme ha aparellat el model i el text. Aquests resultats es recullen a la taula 5.3.

A l'annex C es poden consultar algunes les traces d'execució generades per l'algorisme[2] per algunes de les parelles de la taula de similitud. No s'inclouen totes per motius d'espai però aquestes es poden consultar en el codi font del projecte sota el directori `logs`.

5.1.4 Conclusions

Com es pot veure, analitzant la taula de similituds per files, veiem que l'algorisme assigna correctament totes, tret d'una, de les parelles model-text, i que el cas en el que s'equivoca, el nivell de confiança era el més baix de tots els casos.

Els dos exemples amb una major puntuació de similitud són [REST] i [HOSP]. Després d'una inspecció manual a càrrec del director del projecte, aquest ha corroborat que els dos models esmentats són els que més s'assemblen als respectius textos.

En general, es pot veure com la similitud de la parella òptima està almenys un ordre de magnitud per sobre de la majoria de les altres parelles de cada fila. Aquest resultat es pot veure quantificat a la taula 5.3, on es pot observar que l'algorisme assigna una puntuació entre 3 i 11 vegades superiors a la parella òptima respecte la mitjana de les altres.

Pel que fa al temps d'execució, es pot observar com no hi ha cap correlació significativa entre la similitud d'una parella model-text i el temps d'execució de l'algoritme. On si que sembla haver-hi una correlació és en la mida dels textos i dels models. Els exemples [REST] i [HOSP] són els dos casos més grans del benchmark amb diferència, i es pot veure com els temps d'execució quan es comparen entre ells són notablement superiors. En aquest benchmark s'han utilitzat models des de 8 tasques fins a més de 16. Es pot veure, doncs, que el temps d'execució creix de manera bastant lleu amb la mida del problema, indicant que no es tracta d'un comportament exponencial.

[2] Aquest text es genera internament i s'exposa a la API pública, però no es mostra a l'usuari en la interfície gràfica, ja que és massa detallat.

5.2 Avaluació automàtica d'un conjunt de models per a un text

5.2.1 Plantejament

Aquest experiment pretén avaluar com de bona és la puntuació que l'algorisme troba a l'hora de comparar un model BPMN i una representació textual. Per a l'experiment s'utilitzaran diversos textos i, per cada text, múltiples BPMN que descriguin el mateix procés que aquest però ho facin amb diferents nivells de qualitat. S'espera que l'algorisme puntuï millor aquelles parelles model-text on el model sigui de major qualitat.

5.2.2 Metodologia

Per a fer l'experiment s'utilitzarà el benchmark de parelles de model BPMN i descripció textual "BPMN for Research" de Camunda[5]. Aquest benchmark prové d'un curs de *Business Process Management* on l'activitat final consistia en, donada una descripció textual en llenguatge natural, generar un model BPMN equivalent. El benchmark consta de 4 exercicis i, per cadascun d'aquests, un text model i entre 35 i 60 models BPMN. Els models del benchmark corresponen a totes les solucions que van lliurar els alumnes durant l'activitat.

Els textos utilitzats corresponen als casos `[DISP]`, `[CRED]`, `[REST]` i `[RECO]` de l'annex . Els models no s'han pogut incloure al treball degut al gran nombre d'aquests, però es poden consultar a [5], sota l'apartat *Results* de cadascun dels exercicis.

Les solucions dels alumnes tenen la propietat de que, al no provenir d'experts, varien en qualitat. Malauradament, no es disposa de les qualificacions que els alumnes van obtenir durant el curs. Aquest fet es compensarà amb una validació manual.

Per cada exercici del benchmark amb un text t i un conjunt de models M, s'executarà l'experiment seguint els següents passos.

- Calcular la puntuació de similitud entre el text t i el model m per cada $m \in M$.
- Ordenar les parelles (t, m) de més a menys puntuació i obtenir les tres millors i les tres pitjors.
- Realitzar un anàlisi manual de la qualitat dels tres millors models i comparar-la amb la qualitat dels tres pitjors models.

El resultat esperat és que els tres millors models de cada exercici siguin de millor qualitat que els tres pitjors models. Per determinar la qualitat dels resultats de l'experiment, es recorrerà a un expert[3] que en faci una avaluació manualment.

5.2.3 Resultats

Per motius d'espai, no es poden incloure tots els models dels alumnes ni les traces d'execució pertinents. En aquest apartat es fa un resum dels resultats a nivell general comentant els diversos problemes trobats en aquests.

[3] Josep Carmona, director d'aquest Treball de Fi de Grau.

Degut a diverses malformacions en alguns dels models, ha estat impossible fer-ne un anàlisi. Aquells models que formaven grafs inconsistents, aproximadament un 10%, s'han descartat de l'experiment.

Després d'una avaluació manual dels tres millors i els tres pitjors resultats segons l'algorisme per cadascun dels exercicis del curs de Camunda, s'han fet les següents observacions.

- En els exercicis `[CRED]` i `[RECO]`, els tres millors models de cada classe no són substancialment millors que els tres suposadament pitjors. Alguns models han obtingut una puntuació inusualment alta tot i estar clarament mal modelats.
- En els exercicis `[REST]` i `[DISP]`, els tres millors models de cada classe són una millor representació del text que els tres pitjors models de la classe.
- Els models amb menys tasques solen puntuar-se generalment millor que els models amb moltes tasques.
- En els casos on millor ha realitzat la puntuació l'algorisme, la majoria de models eren estructuralment més similars[4].
- Les puntuacions són, en general, independents del nombre de swimlanes i pools en la resposta dels alumnes.

5.2.4 Conclusions

Es pot dir que l'algorisme ha puntuat correctament els exercicis on el nivell de qualitat era, en general, major. Tot i això, el fet d'haver utilitzar models de menor qualitat ha posat en manifest algunes mancances amb l'enfoc de a l'algorisme que no s'havien previst inicialment. Aquesta falta de previsió prové del fet que s'havia assumit una certa estructura del model que no tots els models compleixen. Tot i que els models que no compleixen aquesta estructura són, en general, de pitjor qualitat, l'algorisme sol assignar-los puntuacions més altes.

Fixant-nos amb més detall amb els casos `[CRED]` i `[RECO]`, on l'algorisme ha fet una pitjor avaluació, s'ha pogut observar que bona part del problema recau en el nombre de tasques. Amb l'algorisme actual, la puntuació és funció de l'assignació òptima de tasques a frases. Un model amb menys tasques, tindrà, en general, una puntuació major. Això es va veure clarament reflexat en el cas concret on l'algorisme assignava la màxima puntuació a un alumne que, segurament per confusió, només va incloure una única tasca al seu model. Com que la única tasca que va incloure contenia literalment les paraules d'una de les frases del model, les úniques característiques generades eren un subconjunt total de les característiques d'una de les frases del text, donant similitud màxima.

Un altre punt que es pot considerar problemàtic amb l'enfoc actual és que l'algorisme no té en compte directament l'estructura de swimlanes i pools en la puntuació. Un model BPMN ha de contenir una swimlane (o pool) per cadascun dels agents involucrats. No obstant això, l'algorisme puntua de forma similar si el model conté swimlanes i pools o si no en té.

Tot això indica una falta de visió global per part de l'algorisme. L'enfoc actual consisteix a aparellar tasques i frases. Les característiques globals només es tenen en

[4] Tot indica que els alumnes van ser guiats d'alguna manera durant els exercicis fent que tots fessin models estructuralment similars

compte indirectament a l'hora de generar el conjunt de característiques d'una tasca o frase concreta. Cal un nou enfoc ampliat que tingués en compte tant l'assignació òptima de tasques a frases com informació a nivell general del text i el model. Un exemple seria considerar la similitud entre el conjunt d'agents en el graf semàntic del text i els agents indicats en les swimlanes i les pools. També es podrien penalitzar solucions on el nombre de tasques i el nombre d'accions esmentades en el text difereixi significativament.

Els resultats d'aquest experiment han resultat molt útils a l'hora de veure quins són els següents passos a l'hora de millorar l'algorisme.

5.3 Efectes de la introducció d'inconsistències

5.3.1 Plantejament

L'objectiu d'aquest experiment és veure com es va modificant el resultat de l'algorisme a mesura que s'introdueixen diferents inconsistències a una parella, inicialment bona, de model BPMN i descripció textual. S'espera que l'algorisme reaccioni adequadament a cadascuna de les inconsistències afegides.

5.3.2 Metodologia

Per a l'experiment s'utilitzarà la parella model-text `[ZOO]` (annex A.10). Es realitzarà la següent seqüència de modificacions al model o al text, i després de cada modificació s'observarà la traça d'execució de la nova assignació calculada per l'algorisme[5]:

1. Primer de tot, s'invertirà l'ordre de les tasques: "Enter information into the system" i "Send request to billing department" al model BPMN. L'objectiu és comprovar que l'algorisme reaccioni a una incoherència amb l'ordre de les tasques i les frases.

2. Mantenint el model modificat, s'eliminarà la frase del text "Once the visitor receives the card, he can co home". Amb aquesta modificació es busca veure què passarà amb les dues tasques que s'assignen inicialment a aquesta frase.

3. Seguint amb el mateix model i text modificats, s'afegirà al model una nova tasca del banc entremig de "Process payment information" i "Charge account" amb el text: "Validate credit card data". Aquesta modificació pretén veure què passaria si aparegués una nova informació al procés de negoci però només s'actualitzés al model BPMN.

4. A continuació, es canviaran de lloc els textos de les swimlanes i les pools. "Visitor" passarà a ser "Marketing department", "ZooClub department" passarà a ser "Bank", "Billing department" passarà a ser "Visitor" i "Bank" passarà a ser "Billing department". Aquest canvi pretén observar com podria l'algorisme detectar una barreja de les swimlanes i les pools causada per un error a l'editor de models o una confusió per part de l'usuari.

[5]Degut a la mida de les traces d'execució, per motius d'espai, no es poden incloure totes en aquesta memòria. No obstant això, els resultats són totalment replicables, ja que el comportament de l'algorisme és determinista.

5.3.3 Resultats

En aquesta secció es descriu el resultat d'aplicar cadascuna de les modificacions de l'apartat anterior sobre l'assignació final. Aquesta és l'assignació inicial que realitza l'algorisme, amb una puntuació de similitud de **0.363**[6]:

Frase	Tasca
"When a visitor wants to become a member of Barcelona's ZooClub, the following steps must be taken."	
"First of all, the customer must decide whether he wants an individual or family membership."	"Decide individual or family ticket"
"If he wants an individual membership, he must prepare his personal information."	"Prepare personal information"
"If he wants a family membership instead, he should prepare the information for its spouse and spawn as well."	"Prepare family's information"
"The customer must then give this information to the ZooClub department."	"Send information to the ZooClub department"
"The ZooClub department introduces the visitor's personal data into the system and takes the payment request to the Billing department."	"Enter information into the system"
"The ZooClub department also forwards the visitor's information to the marketing department."	"Forward information to Marketind department"
"The billing department sends the payment request to the bank."	"Send request to billing department" i "Send payment request"
"The bank processes the payment information and, if everything is correct, charges the payment into user's account."	"Process payment information" i "Charge account"
"Once the payment is confirmed, the ZooClub department can print the card and deliver it to the visitor."	"Deliver ZooClub Card" i "Wait for payment"
"In the meantime, the Marketing department makes a request to mail the Zoo Club's magazine to the visitor's home."	"Mail ZooClub Magazine"
"Once the visitor receives the card, he can go home."	"Wait for card" i "Go home"

Aquesta assignació no és perfecta. En concret, l'algorisme es confon a l'assignar incorrectament la tasca "Send request to billing department" a la frasee "The billing department sends the payment request to the bank". Aquesta confusió prové de la quantitat de paraules similars entre la frase i la tasca en qüestió. A més, l'algorisme detecta que tant a la tasca com a la frase el subjecte és un departament, tot i que no el correcte[7]. Les demés assignacions són correctes.

[6] És important notar que tot i que l'índex d'Overlapping dóna resultats numèrics intuïtius, generalment els valors de similitud es mouen entre 0 i 0.5, sent molt difícil pujar la similitud a partir d'aquest punt. Així doncs, podem considerar que aquesta és una similitud alta.

[7] Això es pot observar amb més detall a la traça d'execució C.4

Després de la primera modificació, aquesta és la nova assignació òptima de l'algorisme, amb una similitud de **0.354**:

Frase	Tasca
"When a visitor wants to become a member of Barcelona's ZooClub, the following steps must be taken."	
"First of all, the customer must decide whether he wants an individual or family membership."	"Decide individual or family ticket"
"If he wants an individual membership, he must prepare his personal information."	"Prepare personal information"
"If he wants a family membership instead, he should prepare the information for its spouse and spawn as well."	"Prepare family's information"
"The customer must then give this information to the ZooClub department."	"Send information to the ZooClub department"
"The ZooClub department introduces the visitor's personal data into the system and takes the payment request to the Billing department."	
"The ZooClub department also forwards the visitor's information to the marketing department."	"Forward information to Marketind department"
"The billing department sends the payment request to the bank."	"Send request to billing department" i "Send payment request"
"The bank processes the payment information and, if everything is correct, charges the payment into user's account."	"Process payment information" i "Charge account"
"Once the payment is confirmed, the ZooClub department can print the card and deliver it to the visitor."	"Deliver ZooClub Card" i "Wait for payment"
"In the meantime, the Marketing department makes a request to mail the Zoo Club's magazine to the visitor's home."	"Mail ZooClub Magazine"
"Once the visitor receives the card, he can go home."	"Wait for card" i "Go home"
No assignada	"Enter information into the system"

Com es pot veure, l'algorisme ha determinat que no hi havia cap assignació vàlida per la tasca "Enter information into the system", perquè assignar-la a la frase que abans era correcta hagués creat un conflicte d'ordre: Dues tasques t_1 i t_2 tals que $t_1 \rightsquigarrow t_2$ no poden anar assignades a dues frases s_1 i s_2 tals que $s_1 \nleftrightsquigarrow s_2$. El sistema identifica correctament aquesta inconsistència i descarta una de les dues frases alertant a l'usuari de que hi ha tasques no assignades.

D'esprés d'eliminar la frase "Once the visitor receives the card, he can go home", aquesta és la nova assignació, amb una puntuació de **0.324**:

Frase	**Tasca**
"When a visitor wants to become a member of Barcelona's ZooClub, the following steps must be taken."	
"First of all, the customer must decide whether he wants an individual or family membership."	"Decide individual or family ticket"
"If he wants an individual membership, he must prepare his personal information."	"Prepare personal information"
"If he wants a family membership instead, he should prepare the information for its spouse and spawn as well."	"Prepare family's information"
"The customer must then give this information to the ZooClub department."	"Send information to the ZooClub department"
"The ZooClub department introduces the visitor's personal data into the system and takes the payment request to the Billing department."	
"The ZooClub department also forwards the visitor's information to the marketing department."	"Forward information to Marketind department"
"The billing department sends the payment request to the bank."	"Send request to billing department" i "Send payment request"
"The bank processes the payment information and, if everything is correct, charges the payment into user's account."	"Process payment information" i "Charge account"
"Once the payment is confirmed, the ZooClub department can print the card and deliver it to the visitor."	"Deliver ZooClub Card", "Wait for payment" i "Wait for card"
"In the meantime, the Marketing department makes a request to mail the Zoo Club's magazine to the visitor's home."	"Mail ZooClub Magazine" i "Go home"
No assignada	"Enter information into the system"

Com es pot veure, la tasca "Wait for card" ha passat a estar assignada a una altra frase, així com "Go home". La puntuació per la tasca "Go Home" s'ha reduït de 0.65 a 0.20 mentre que la de de "Wait for card", en comparació, s'ha mantingut aproximadament amb el mateix valor.

En afegir la nova tasca "Validate credit card data", aquesta és la nova assignació, amb una puntuació de **0.306**:

Frase	Tasca
"When a visitor wants to become a member of Barcelona's ZooClub, the following steps must be taken."	
"First of all, the customer must decide whether he wants an individual or family membership."	"Decide individual or family ticket"
"If he wants an individual membership, he must prepare his personal information."	"Prepare personal information"
"If he wants a family membership instead, he should prepare the information for its spouse and spawn as well."	"Prepare family's information"
"The customer must then give this information to the ZooClub department."	"Send information to the ZooClub department"
"The ZooClub department introduces the visitor's personal data into the system and takes the payment request to the Billing department."	
"The ZooClub department also forwards the visitor's information to the marketing department."	"Forward information to Marketind department"
"The billing department sends the payment request to the bank."	"Send request to billing department" i "Send payment request"
"The bank processes the payment information and, if everything is correct, charges the payment into user's account."	"Process payment information" i "Charge account"
"Once the payment is confirmed, the ZooClub department can print the card and deliver it to the visitor."	"Deliver ZooClub Card", "Wait for payment" i "Wait for card"
"In the meantime, the Marketing department makes a request to mail the Zoo Club's magazine to the visitor's home."	"Mail ZooClub Magazine" i "Go home"
No assignada	"Enter information into the system" i "Validate credit card data"

Es pot observar com l'algorisme no pot assignar la nova tasca a cap frase amb una puntuació de similitud prou alta.

Finalment, després de barrejar els textos de les swimlanes i les pools, l'assignació s'ha mantingut igual. No obstant això, la puntuació de similitud global ha disminuït fins a **0.276**.

5.3.4 Conclusions

En aquest experiment s'han realitzat diverses modificacions, cadascuna empitjorant cada vegada més una parella model-text inicialment bona. En quant a la puntuació, es pot observar que aquesta ha anat disminuint consistentment després de cadascuna de les modificacions. La similitud s'ha reduït en total un 24% entre l'assignació inicial i la quarta modificació. Podem dir, doncs, que la puntuació de similitud es pot utilitzar com a indicador d'inconsistències. Després d'introduir una modificació al model i al text, si es detecta una baixada en la similitud, és molt probable que la causa sigui una

inconsistència entre els dos.

A la primera modificació, es pot veure com l'algorisme aplica correctament les restriccions d'ordre. A més, el canvi d'ordre entre dues tasques té un impacte molt negatiu a la similitud.

Pel que fa a la segona modificació, s'ha pogut observar com l'algorisme reacciona a l'eliminació d'una frase assignant les tasques a noves frases que no tenen sentit, disminuint considerablement la similitud de cadascuna d'aquestes dues tasques. En un cas ideal, la similitud hauria de disminuir tant que quedaria sota del llindar d'assignació. No obstant això, el llindar òptim depèn de cada cas i és molt difícil determinar automàticament un llindar perfecte sense moltes més dades d'exemple de les que es disposa. Davant una modificació d'aquest tipus, després d'inspeccionar l'assignació òptima, l'usuari pot detectar sense problemes que hi ha tasques que s'estan assignant on no toca perquè no existeix la frase on haurien d'anar realment assignades.

La tercera modificació, afegir una nova tasca al model però no fer-ho al text, genera un comportament molt predictible. La nova tasca s'assigna a una frase sense cap mena de sentit, simplement pel fet que algunes de les paraules s'assemblen, i amb una disminució dràstica de la similitud. Davant aquest fet, seria senzill arreglar el problema reflectint l'acció "Go home" explícitament al text. Si no existís cap frase amb paraules similars a les de la frase eliminada, el comportament seria encara més intuïtiu, ja que les tasques quedarien sense assignar.

Després de barrejar les pools i les swimlanes, l'algorisme manté l'assignació, però baixa la similitud global. Això és així, perquè els subjectes no són la única cosa que l'algorisme utilitza per guiar la seva execució. En concret, amb el conjunt de pesos de característiques actuals, val exactament el mateix que una frase i una tasca comparteixin agent o que comparteixin objecte directe[8]. En aquest model, totes les tasques i les frases indiquen clarament tant l'agent com el pacient, o objecte, de l'acció. En barrejar els textos de les pools i les swimlanes, els agents deixen de coincidir entre les frases i les tasques, però l'algorisme és capaç d'inferir la mateixa assignació a partir de tota l'altra informació. En un escenari com aquest, és difícil sense una inspecció en detall, veure on està el problema, però la baixada de puntuació és un indicador clar de que alguna cosa ha empitjorat.

Resumint, d'aquest experiment se'n poden extreure dues conclusions fonamentals. En primer lloc, la importància del llindar. Determinar un llindar òptim pel cas concret ajudarà a que les reaccions de l'algorisme siguin més intuïtives, marcant les tasques que no apareguin al text com a tal, i no assignant-les a una frase sense massa sentit. Per altra banda, cal destacar la robustesa general de l'algorisme. Després d'aplicar modificacions molt precises i concretes a l'algoritme, les reaccions han afectat només a la part rellevant de l'assignació, mantenint exactament igual la resta de les altres assignacions.

[8] El conjunt de pesos de l'algorisme és totalment configurable. Si un usuari prefereix donar més pes als agents, simplement ha de modificar-ne el pes.

Capítol 6

Gestió del projecte

6.1 Planificació temporal

Per un bon desenvolupament del projecte és indispensable fer una previsió de les tasques que es realitzaràn i el temps que s'invertirà en aquestes. Una bona planificació ha de ser flexible, i tenir en compte possibles imprevistos i desviacions que es puguin originar. En aquesta secció es descriu la planificació temporal realitzada per al projecte.

6.1.1 Descripció de les tasques

En aquesta secció es descriuen les diferents tasques que s'han dut a terme des de l'inici fins a la finalització del projecte.

Disseny de l'algorisme

La tasca de disseny de l'algorisme consisteix a decidir com s'implementaran les diferents parts d'aquest. A diferència de les altres tasques és difícil de subdividir en seccions menors. Això és així perquè no se sap sobre quines parts de l'algorisme s'han de prendre decisions fins que no avança prou la implementació d'aquest. Aquesta part del projecte és un procés exploratori on s'avaluen diverses alternatives i possibles idees per incorporar al programa.

Aquesta tasca es durà a terme en reunions setmanals amb l'alumne i els directors del projecte. En aquestes reunions es prendran conjuntament les diferents decisions sobre l'algorisme en relació a nous fets que hagin aparegut fruit de l'experimentació o de l'avanç en la implementació de l'algorisme.

Configuració del sistema i l'entorn de programació

Prèviamen a la tasca d'implementació s'han d'instal·lar i configurar tots els paquets de software necessaris per a aquest projecte. Una llista d'aquests es pot trobar a la secció 6.1.3. També s'ha de configurar un entorn de programació que permeti executar i compilar codi tant en *Clojure* com *Java*. Cal finalment configurar el projecte amb *Maven* per poder gestionar fàcilment les dependències de llibreries per ambdós llenguatges durant

tota la durada del projecte i per facilitar l'integració del codi en el servidor.

Implementació de l'algorisme

Aquesta és una de les fases principals que s'estén durant pràcticament tota la durada del projecte. Consisteix en crear el codi que executarà l'algorisme i realitzar les proves necessaries per determinar-ne el correcte funcionament. La major part d'aquesta tasca s'ha dut a terme utilitzant *Clojure*, tret de les parts de l'interfície pública que utilitzarà el servidor per cridar l'algorisme, que s'han programat utilitzant *Java*.

La tasca de programació de l'algorisme es divideix, a grans trets, en els diferents mòduls descrits al capítol 4. En la majoria de casos s'han pogut desenvolupar els diferents mòduls en paral·lel, degut a que la seva funcionalitat quedava desacoplada. Si no s'indica res és que la implementació d'aquest mòdul està deslligada de les altres. A continuació es defineixen amb detall cadascuna de les parts a implementar:

Extracció de característiques: Consisteix a convertir les frases del text i les tasques del model en vectors plans de característiques (*features*). Per l'extracció del text aquesta part usarà *Freeling* extensivament. D'altra banda, per l'anàlisi del model s'usarà *Freeling* per una banda i la llibreria *Activiti* per l'altra.

Càlcul de similaritats: Consisteix a implementar una mètrica que permeti calcular la similaritat entre vectors de característiques generats pel mòdul extractor.

Càlcul d'ordre: Consisteix a implementar el càlcul d'una noció de distància entre frases del text i tasques del model. L'objectiu d'aquest mòdul és que donats el text i el model es pugui establir un ordre parcial entre elements d'un i de l'altre.

Solver: Consisteix a utilitzar un algorisme ja implementat d'optimització global o satisfacció de restriccions que resolgui el problema de matching de frases a tasques. Requereix que tots els mòduls anteriors estiguin funcionant correctament.

Integració amb el servidor: Crear una interfície pública per executar l'algorisme de forma senzilla des del servidor web que inclou aquest algorisme i assistir amb els errors que aquesta integració pugui ocasionar.

Experimentació

Les tasques d'experimentació es duen a terme després de finalitzar la implementació d'un dels mòduls principals del projecte. L'objectiu és avaluar el rendiment de les diferents parts de l'algorisme per detectar en quins punts es podria millorar. Podem veure, doncs, la tasca d'experimentació com una tasca que es desenvoluparà en paral·lel a la implementació.

No es pot definir quins experiments s'han de realitzar a priori, però una aproximació raonable és assumir que caldrà realitzar experiments per validar cadascun dels mòduls que s'han de programar a la tasca d'implementació.

Cal notar que dins d'aquesta tasca també entrarien els experiments finals inclosos en aquesta memòria, però aquests no han estat els únics que s'han realitzat.

Documentació

Aquesta fase engloba tot el relacionat amb l'elaboració de la documentació del projecte. Se'n poden distingir diverses parts:

Lliurables de GEP: Elaborar cadascun dels lliurables de l'assignatura de GEP.

Memòria del projecte: Elaborar el document final de la memòria del projecte. Comença just després d'acabar GEP.

Presentacions orals: Preparar el suport visual, redactar el guió i realitzar els assajos pertinents per la lectura final del projecte. Es realitzarà al final del projecte, després de l'entrega de la memòria.

6.1.2 Taula de temps

La taula 6.1 resumeix els temps esperats per a completar cadascuna de les tasques del projecte:

Tasca	Temps (h)
Disseny de l'algorisme	**90**
Configuració del sistema i entorn de programació	**15**
Implementació de l'algorisme	**240**
... Extractors de característiques	70
... Càlcul de similaritats	60
... Càlcul d'ordre	30
... Solver	70
... Integració amb el servidor	10
Experimentació	**85**
Documentació	**100**
... Lliurables de GEP	25
... Memòria del projecte	50
... Presentacions orals	25
Total	**530**

Taula 6.1: Temps de realització esperats per cadascuna de les tasques.

Canvis en la planificació de tasques

Respecte la fita inicial d'aquest projecte la previsió de tasques s'ha vist modificada. En concret, dins la tasca d'implementació s'ha eliminat el desenvolupament del LAP Solver. Els temps d'aquesta part s'han inclòs dins la tasca del Solver, ja que l'estructura de l'algorisme finalment no ha inclòs el LAP Solver i no té sentit considerar-lo com una tasca d'implementació separada. D'altra banda, s'ha eliminat la secció de detecció d'inconsistències. Això és així perquè aquesta s'ha realitzat finalment de manera implícita, fent que no s'hagués de desenvolupar com una tasca pròpiament dita, quedant repartida entre diverses parts. Finalment, s'ha afegit una tasca d'integració del codi en el servidor web. Aquesta tasca no s'havia previst inicialment ja que es va considerar que el cost en hores seria menys significatiu del que ha estat. Tot això ha comportat una variació en els temps de les diferents parts: Per una banda s'ha produit un augment significatiu

d'hores en la tasca d'implementació del solver, degut a que va caldre implementar més solucions de les previstes fins que es va trobar la bona.

6.1.3 Recursos utilitzats

En aquesta secció es descriuen els diversos recursos que s'utilitzaran per a duur a terme les tasques planejades en a aquest projecte. Estan dividits en dues seccions: recursos de *software* i de *hardware*.

Recursos de Hardware

Ordinador portàtil: Utilitzat per a desenvolupar l'algorisme i, elaborar la documentació del projecte. Les seves especificacions tècniques són: Intel Core i7-3630QM a 2.40GHz, 8GB de memòria RAM, consum de 120W

Màquines del *Textserver*: Aquest projecte utilitza *Freeling* com a software d'anàlisi de llenguatge natural. Com que els algorismes que implementa freeling són bastant costosos l'algorisme envia els textos a aquest *webservice* que executa *Freeling*. El *Textserver* s'executa al *cluster* del departament de CS de la FIB, amb més de 160 màquines i 1000 nuclis de CPU.

Recursos de Software

Freeling: Software d'anàlisi i processament de llenguatge natural. Utilitzat per la part d'anàlisi de textos de l'algorisme.

Activiti: Llibreria de Java per a generar, analitzar i executar models BPMN. També disposa d'un editor i visualitzador de models.

LPSolve: Llibreria de resolució de problemes d'*Integer Linear Programming*.

Arch GNU/Linux: El sistema operatiu utilitzat per a realitzar aquest projecte.

Clojure: Llenguatge de programació amb el que s'implementa l'algorisme. És un dialecte de LISP funcional i dinàmic compatible amb la JVM, i per tant, Java.

Java: L'altre llenguatge de programació amb el que s'implementa l'algorisme. S'usa per generar una API externa del projecte.

Maven Software per a gestió de projectes basats en Java. Usat en la gestió de dependències en el projecte.

LaTeX: Paquet de software pel maquetat de documents. S'utilitza en l'elaboració de la documentació del projecte. Es fa servir tant via la plataforma web ShareLaTeX com en local en funció de les necessitats.

Gantter: Eina de creació de diagrames de Gantt. Per a la planificació temporal del projecte.

6.1.4 Diagrama de Gantt

En aquesta secció s'inclouen dos diagrames de Gantt, el de la figura 6.1 és el diagrama de la planificació del projecte a priori. El diagrama de la figura 6.2 és el diagrama actualitzat considerant els canvis en la planificació temporal, és a dir, reflexa els temps reals invertits en cadascuna de les parts del projecte.

Com es pot veure, el major canvi ha estat que l'integració amb el servidor ha durat bona part del projecte, tot i durar poques hores. També s'han vist modificada la durada de la tasca del càlcul d'ordre, degut a l'imprevist que va suposar modelar incorrectament l'ordre inicialment i l'integració, totalment imprevista, de l'algorisme de behavioral profiles que això va suposar.

	Name	Duration	Start	Finish	Predecessors
1	Disseny de l'algoritme	108d?	15/01/2016	15/06/2016	
2	Configuració del sistema i entorn de programaci	6d?	20/01/2016	27/01/2016	
3	**Implementació**	**92d?**	**28/01/2016**	**03/06/2016**	
4	Extractors de característiques	67d?	28/01/2016	29/04/2016	2
5	Càlcul de similaritats	64d?	28/01/2016	26/04/2016	2
6	Càlcul d'ordre	44d?	29/02/2016	28/04/2016	2
7	LAP Solver	16d?	29/02/2016	21/03/2016	2
8	Solver	14d?	02/05/2016	19/05/2016	4,5,6,7
9	Detecció d'incoherències	11d?	20/05/2016	03/06/2016	8
10	Experimentació	96.5d?	28/01/2016	10/06/2016	4SS
11	**Documentació**	**90d?**	**22/02/2016**	**24/06/2016**	
12	Lliurables de GEP	28d?	22/02/2016	30/03/2016	
13	Memòria del projecte	57d?	31/03/2016	17/06/2016	12
14	Presentació oral	5d?	20/06/2016	24/06/2016	1,9,13

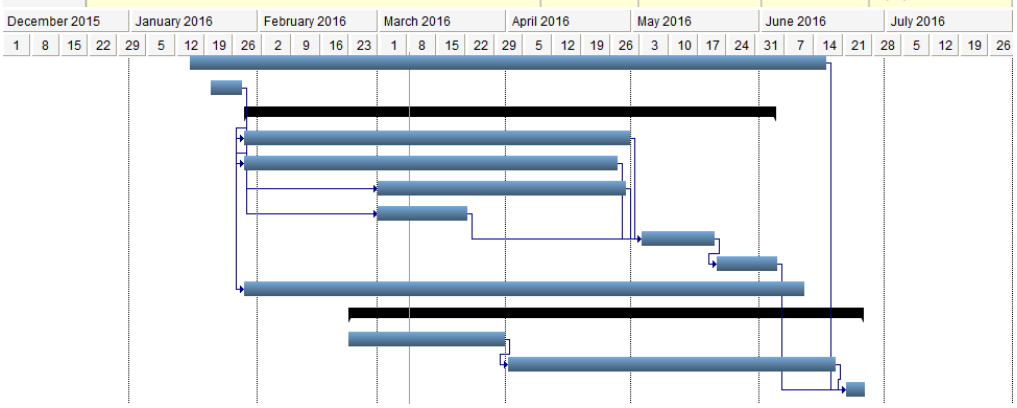


Figura 6.1: El diagrama de gantt de la planificació temporal del projecte abans de la realització.

	Name	Duration	Start	Finish	Predecessors
1	Disseny de l'algoritme	108d?	15/01/2016	15/06/2016	
2	Configuració del sistema i entorn de programació	6d?	20/01/2016	27/01/2016	
3	**Implementació**	97d?	**28/01/2016**	**10/06/2016**	
4	Extractors de característiques	52d?	28/01/2016	08/04/2016	2
5	Càlcul de similaritats	67d?	28/01/2016	29/04/2016	2
6	Càlcul d'ordre	50d?	29/02/2016	06/05/2016	2
7	Integració amb el servidor	73.5d?	29/02/2016	09/06/2016	2
8	Solver	14d?	09/05/2016	26/05/2016	4,5,6
9	Detecció d'incoherències	11d?	27/05/2016	10/06/2016	8
10	Experimentació	96.5d?	28/01/2016	10/06/2016	4SS
11	**Documentació**	90d?	**22/02/2016**	**24/06/2016**	
12	Lliurables de GEP	28d?	22/02/2016	30/03/2016	
13	Memòria del projecte	57d?	31/03/2016	17/06/2016	12
14	Presentació oral	5d?	20/06/2016	24/06/2016	1,9,13

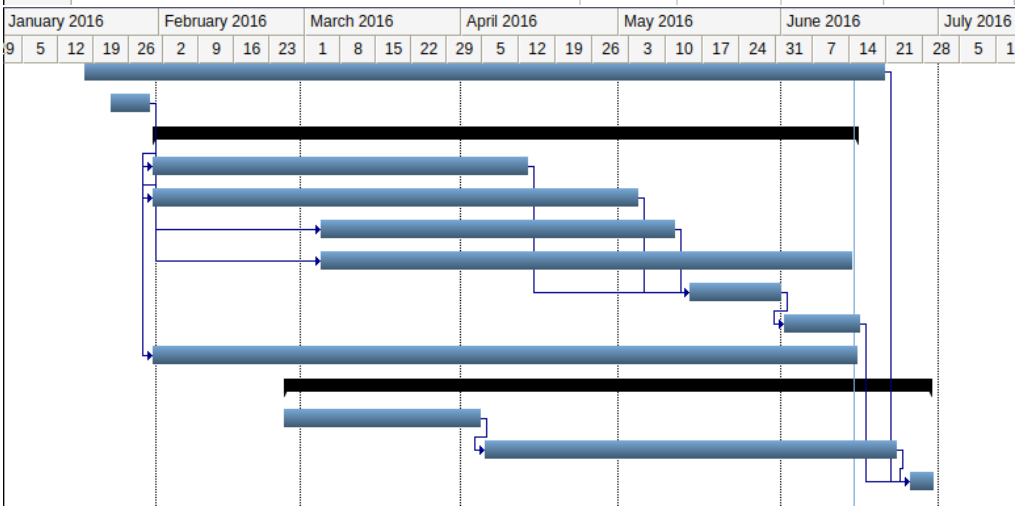


Figura 6.2: El diagrama de gantt que reflexa l'evolució real que ha tingut el projecte.

6.1.5 Pla d'acció inicial i desviacions

El pla d'acció inicial ha consistit en duur a terme les tasques planejades tal com estaven especificades al diagrama de Gantt de la fita inicial. No obstant això, s'ha de tenir en compte que aquest és un projecte amb un fort component exploratori. Així doncs, a priori ja es va detectar que podien aparèixer problemes per adaptar-se al pla dissenyat. En qualsevol moment una decisió de disseny provinent de nous fets fruit de l'experimentació pot fer que algun dels mòduls de l'algorisme deixi de ser necessari o necessiti canvis substancials. Això ha fet que la planificació inicial anés evolucionant al llarg del projecte.

L'objectiu principal del projecte és molt clar: Tenir un algorisme que funcioni per a resoldre el problema descrit. Tot i això inicialment hi havia moltes parts en les quals hi havia molt poc decidit i on resultava difícil fer una estimació raonable d'hores. L'eina principal de control de temps ha estat la quantitat d'exploració en les diferents parts de l'algorisme.

El primer lloc on s'ha trobat molta flexibilitat és en la quantitat de característiques que s'extreuen del text i el model. Com més tipus de característiques s'exploren més qualitat té l'algorisme[1]. Inicialment s'havia plantejat que, en cas de falta de temps es podria optar per explorar menys tipus de característiques a extreure. Finalment, el conjunt de característiques que s'han explorat ha estat prou gran, i dóna uns resultats bastant satisfactoris.

Un altre punt que es va detectar que podia ajudar a ajustar el temps final de desenvolupament és la quantitat d'experimentació. Com més alternatives s'explorin pel que fa a mètriques de similaritat i algorismes d'optimització millor serà l'algorisme final. El punt on s'ha invertit més temps d'experimentació ha estat en el càlcul de l'assignació òptima, la tasca del Solver. Això ha fet que s'hagués de retallar temps de l'exploració de diferents mètriques de similaritat.

Finalment, la tasca d'intentar reduir el problema a un LAP, per veure si la solució obtinguda és una bona aproximació, s'havia considerat com opcional. No obstant això finalment s'ha pogut explorar l'opció i implementar-la.

Com es pot veure a la taula de temps. La quantitat d'hores prevista per completar aquest projecte és de 530 hores, que coincideix amb les hores per crèdit orientatives establertes a la normativa del treball de final de grau a la FIB. Tenint en compte que la durada del projecte ha estat de 24 setmanes aproximadament, podem veure que la dedicació mitjana ha estat de 22 hores setmanals, que descomptant dos dies per setmana equival a 4,4 hores de treball diàries. Aquesta és una quantitat d'hores viable.

6.2 Pressupost

En un projecte real és molt important fer un anàlisi previ dels costos d'un projecte. L'estimació d'un pressupost és indispensable de cara a convèncer potencials inversors, per exemple. En aquesta secció es fa un anàlisi detallat dels costos per a la realització d'aquest TFG. Cal tenir en compte que aquest és un pressupost fictici que s'ha dissenyat com si es tractés d'un projecte d'enginyeria en una empresa real, i que no tots els costos –Especialment els referents als recursos humans– reflecteixen la realitat del projecte.

[1] Aixo no vol dir que l'algorisme final faci l'extracció de totes les característiques implementades. Ans al contrari, com més d'aquestes es plantegin i s'avaluin més informat serà l'algorisme i per tant, en general, millor.

6.2.1 Recursos humans

La major part de tasques d'aquest projecte corresponen a una sola persona[2]. Aquest, però, no seria el cas en un projecte real. A l'hora de fer aquest pressupost s'han tingut en compte els diferents rols que intervindrien en un projecte similar. Així doncs, els recursos humans s'han dividit en els rols de: Cap de projecte, dissenyador, programador, *beta tester* i documentador.

Tenint en compte les diferents tasques definides a la planificació temporal, podem estimar les hores de cadascun dels membres de l'equip:

- La configuració de l'entorn de programació correspondria al programador, que és qui estableix els requisits de l'entorn. (15h)
- El disseny de l'algorisme correspondria conjuntament al Cap de projecte i al Dissenyador. El temps planejat de la part de disseny no es parteix, és simultani[3]. (90+90h)
- La implementació de l'algorisme corresponen al programador i al *beta tester*. El temps total es reparteix entre els dos rols. (170+70h)
- L'experimentació correspon al dissenyador, encarregat de dissenyar els experiments. Es comptabilitzen també unes hores del programador per implementar el codi dels experiments. (75+10h)
- La documentació correspon al rol de documentador. (100h)

A la taula 6.2 es pot veure el pressupost estimat considerant un sou adequat per cadascun dels rols.

Rol	Preu per hora	Hores	Cost
Cap de projecte	50.00€/h	90h	4500.00€
Dissenyador	35.00€/h	165h	5775.00€
Programador	30.00€/h	180h	5400.00€
Beta tester	30.00€/h	70h	2100.00€
Documentador	30.00€/h	100h	3000.00€
TOTAL			20775.00€

Taula 6.2: Estimació del pressupost dels recursos humans.

6.2.2 Recursos de hardware

A la taula 6.3 es descriu el pressupost que s'ha estimat per als recursos de hardware necessaris per dur a terme el projecte. La falta de dades sobre les màquines del *Textserver* fa que no en puguem estimar un pressupost. D'altra banda res impedeix instal·lar *Freeling* en local i executar-lo en la mateixa màquina amb la qual es desenvolupa el projecte si el servidor suposés un cost important.

[2] Tret de la tasca de disseny de l'algorisme, que es realitza de manera conjunta entre l'alumne i els directors de projecte.

[3] És important notar aquest fet, ja que fa que les hores del pressupost no sumin el mateix que les hores totals del projecte.

Producte	Preu	Vida útil	Amortització
Ordinador portàtil	700.00€	4 anys	50.00€
Màquines del *Textserver*	0.00€	desconegut	0.00€
TOTAL	700.00€		50.00€

Taula 6.3: Estimació del pressupost dels recursos de software.

6.2.3 Recursos de software

A la taula 6.4 es descriu el pressupost estimat per als recursos de software del projecte. Com es pot observar el pressupost és purament un formalisme, ja que tots els recursos de software utilitzats són totalment gratuïts.

Producte	Preu	Vida útil	Amortització
Freeling	0.00€	–	0.00€
Activiti	0.00€	–	0.00€
LPSolve	0.00€	–	0.00€
Arch GNU/Linux	0.00€	–	0.00€
Clojure	0.00€	–	0.00€
Java	0.00€	–	0.00€
Maven	0.00€	–	0.00€
LaTeX	0.00€	–	0.00€
Gannter	0.00€	–	0.00€
TOTAL			0.00€

Taula 6.4: Estimació del pressupost dels recursos de software.

6.2.4 Despeses indirectes

Les despeses indirectes tenen en compte l'ús d'electricitat i material d'oficina necessaris per a la realització d'aquest projecte. La taula 6.5 mostra una estimació del pressupost d'aquestes despeses.

El preu de l'electricitat s'ha calculat tenint en compte el preu del KW/h a la ciutat de Barcelona i les hores del projecte en les que, segons la planificació, és necessari l'ús de l'ordinador[4]. A més, s'inclou una part addicional en previsió de la llum necessària per desenvolupar el projecte.

Pel que fa al material d'oficina s'ha escollit un preu raonable per tal que la falta de material no resulti un inconvenient. Cal destacar que aquest projecte no en requereix un ús intensiu.

Producte	Preu	Unitats	Cost
Electricitat (portatil)	0.141€/KWh	54Kwh[5]	7.61€
Electricitat (llum)	0.141€/KWh	20Kwh[6]	7.61€
Material d'oficina	25.00€/unitat	1	25.00€
TOTAL			32.61€

Taula 6.5: Estimació del pressupost dels recursos indirectes.

[4]L'ordinador consumeix 120W, com es comenta a l'apartat de recursos.

6.2.5 Pressupost total

A la taula 6.6 es mostra el pressupost global del projecte, on es pot veure que la part que aporta més al cost del projecte és la dels recursos humans. A l'apartat 6.3.1 es realitza la valoració global d'aquest pressupost.

Concepte	Preu
Recursos Humans	20775.00€
Recursos de Software	0.00€
Recursos de Hardware	50.00€
Despeses indirectes	32.61€
TOTAL	20857.61€

Taula 6.6: Estimació del pressupost total.

6.2.6 Control de gestió

En aquesta secció es considera com podria modificar-se aquest pressupost davant d'imprevistos i es proposen alternatives per tal de compensar-ho.

Com es veu a l'apartat anterior, pràcticament la totalitat del cost del projecte correspon als recursos humans. És per aquest fet que possibles desviacions en la planificació temporal poden afectar molt significativament al pressupost del projecte. Si es dóna el cas, cal sospesar si és més profitós reduir funcionalitats del projecte final o augmentar les hores totals del projecte. El primer escenari possiblement comportarà menys beneficis donat que s'haurà obtingut un producte de menys qualitat, mentre que amb el segon pot augmentar considerablement el pressupost final.

Pel que fa al software, un possible imprevist seria la incorporació de codi amb llicències de pagament en el projecte. En cas de ser inevitable s'haurà d'assumir el cost però sempre que sigui possible s'optarà per alternatives gratuïtes i, si és possible, de codi obert.

Pel que fa al hardware i al consum elèctric no es preveuen desviacions. Realment l'únic necessari per implementar l'algorisme és un ordinador i algun material complementari (papers, bolígrafs, etc.).

6.3 Informe de sostenibilitat

La dimensió econòmica no és la única que s'ha d'analitzar per a assegurar la viabilitat d'un projecte. Les altres dues dimensions de la sostenibilitat, la ambiental i la social, són tant o més importants que aquesta. En aquesta secció es realitza l'anàlisi de sostenibilitat feta per aquest projecte. En primer lloc es presenta la matriu de sostenibilitat i posteriorment se'n realitza una anàlisi per files justificant les puntuacions assignades a cada casella d'aquesta.

6.3.1 Matriu de sostenibilitat

La taula 6.7 mostra la matriu de sostenibilitat d'aquest projecte. La puntuació de sostenibilitat total del projecte és de 70 punts, el que ens indica que és un projecte

suficientment viable i sostenible.

	PPP	Vida útil	Riscos
Ambiental	10	19	0
Econòmic	8	15	-12
Social	10	8	-5
Puntuació	28	42	-17
Total	70		

Taula 6.7: Matriu de sostenibilitat

Dimensió ambiental

Aquest projecte no requereix l'ús de manera directa de cap agent contaminant. L'únic necessari per al desenvolupament de l'algorisme és un ordinador, i el producte final és software. Tot i això l'ús d'energia elèctrica i aparells electrònics comporta, indirectament, l'emissió de gasos contaminants a l'atmosfera. Fent una estimació[7] del CO_2 emès, es calcula que el desenvolupament d'aquest projecte comportarà l'emissió de 48.1Kg de CO_2 alliberats a l'atmosfera. És important destacar que la utilització d'un ordinador portàtil com a eina principal de desenvolupament fa que el consum energètic sigui considerablement inferior que si s'hagués utilitzat un ordinador de sobretaula[8]. En tot cas, la quantitat de CO_2 generada per aquest projecte és molt inferior a la que genera una persona en els 6 mesos de durada d'aquest. Per aquest fet, s'ha donat la puntuació màxima de 10 punts a la casella Ambiental/PPP de la matriu.

Si aquest projecte s'acaba convertint en un software exitós, és possible que moltes empreses l'incorporin en els seus servidors i l'utilitzin de manera extensiva. És difícil estimar el consum energètic que això comportarà, però és innegable que un software executant-se contínuament en els servidors d'una empresa i que fa servir algorismes costosos té un cost energètic no negligible. D'altra banda, aquest fet es veuria possiblement compensat per l'estalvi de paper i material que suposaria l'automatització de totes les tasques relacionades amb el *Business Process Management*. És per aquest fet que s'ha donat una puntuació de 19 a la casella Ambiental/Vida útil de la matriu de sostenibilitat.

Finalment, cal comentar que no es preveu cap escenari en el qual aquest projecte pugui augmentar la seva petjada ecològica més enllà del que ja s'ha comentat en aquest document. És per això que s'assigna una puntuació de 0 a la casella Ambiental/Riscos.

Dimensió econòmica

A l'apartat 6.2 ja es realitza una anàlisi detallat de tots els costos que intervenen en aquest projecte, tant humans com materials. Cal tenir en compte, però, que aquest és le pressupost de realització del projecte l'objectiu del qual és crear un software. El software resultant molt probablement requereixi manteniment, actualitzacions i expansions.

És difícil reduir costos durant la realització del projecte. Quasi tot el cost del projecte recau en el cost dels sous dels desenvolupadors i les hores vénen marcades per

[7] S'ha utilitzat la calculadora d'emissions de CO2 disponible a: `http://arboliza.es/compensar-co2/calculo-co2.html`

[8] Un ordinador de sobretaula equivalent consumiria uns 400W de mitjana en comptes dels 120W que gasta el portàtil.

la planificació temporal. No obstant una planificació més compacta i no tan centrada en l'experimentació podria ajudar a millorar en l'àmbit econòmic. El fet que aquest projecte aborda un problema que gairebé no s'ha tractat fins al moment a la literatura fa que sigui molt difícil parlar de reaprofitar codi d'altres parts per l'algorisme. A part, el cost tant en software com hardware és el mínim necessari. Per tot això s'assigna una puntuació de 8 a la casella Econòmic/PPP.

L'objectiu del projecte global és comercialitzar el producte resultant a empreses[9]. La viabilitat econòmica d'aquest paquet de software depèn del model de distribució que s'esculli per comercialitzar-lo. Per una banda es pot optar per una llicència tancada i de pagament. Una possible alternativa a la venda per llicència, és un model de doble llicència: S'ofereix una versió bàsica del producte com a software gratuït perquè qualsevol empresa el pugui provar i es cobra per la versió completa, que permet l'ús del software per a mitjans comercials. És difícil determinar a priori el potencial de vendes del producte resultant, però si l'algorisme genera bons resultats, pot arribar a ser una eina de gran valor per a empreses centrades en el BPM. És per això que s'ha donat una puntuació de 16 a la casella Econòmic/Vida útil.

Si l'enfoc del tractament automàtic del *Business Process Management* resulta profitós a les empreses centrades en aquesta pràctica, és molt probable que apareguin tot tipus de competidors al software d'aquest projecte. No obstant cal tenir en compte que s'estaria partint d'una posició avantatjosa, ja que aquest seria el primer software al mercat a oferir funcionalitats similars. És important tenir en compte aquest risc a l'hora de comercialitzar el software final i per això s'ha donat una puntuació de -12 a la casella Econòmic/Riscos.

Dimensió social

Aquesta és, potser, la dimensió més rellevant per aquest projecte. Tant en aspectes positius com també en negatius.

Pels actors implicats directament en el desenvolupament, el procés comporta un creixement personal. Concretament, com a alumne, m'acosta al món de la recerca, i em dóna una bona visió de l'algorítmia en un entorn pràctic i realista com és el *Business Process Management* en combinació amb el *Natural Language Processing*. A part, el fort component de programació del projecte i l'elecció de les eines de treball em comporta un creixement personal especialment com a programador. A més d'haver treballat en un projecte gran i amb utilitat pràctica directa per primera vegada. És per això que s'ha puntuat amb un 10 la casella Social/PPP de la matriu.

La documentació i anàlisi dels processos de negoci en empreses resulta una manera molt bona de controlar les activitats d'una empresa de cara a optimitzar-ne el cost. És evident, doncs, que l'automatització d'aquestes tasques tindrà un impacte positiu directe en el rendiment econòmic de l'empresa. A més, l'algorisme desenvolupat en aquest TFG pot estalviar una gran quantitat d'hores de treball repetitiu assistint en la cerca d'incoherències entre els models BPMN i les seves representacions textuals. No obstant això, aquest mateix estalvi pot resultar en un impacte negatiu pel col·lectiu d'empleats que treballi desenvolupant aquestes mateixes tasques. Aquest últim fet es veuria especialment accentuat si les empreses decideixen retallar en la seva plantilla com a resultat de l'automatització d'aquesta tasca. És per això que s'assigna una puntuació

[9]Cal recordar que l'objectiu d'aquest TFG és l'algorisme que es desenvolupa en aquest TFG passi a formar part d'un paquet software que pugui servir per a les empreses centrades en *Business Process Management*. Em refereixo a això amb el projecte global.

de 15 a la casella Social/Vida útil de la matriu de sostenibilitat.

Com ja s'ha comentat, la pèrdua de llocs de treball fruit de l'automatització de les tasques en les quals assisteix l'algorisme d'aquest projecte pot ser vista com un risc per a la gent que treballa en aquest tipus de feines. No obstant s'ha de tenir en compte que això també pot ser vist com una oportunitat de reassignar aquests treballadors en tasques de més alt nivell. A més, si el resultat d'aquest projecte comporta beneficis a una empresa aquesta té un major potencial de creació de llocs de treball. Hi han riscos, però es poden evitar. Per això s'assigna una puntuació de -5 a la casella Social/Riscos de la matriu de sostenibilitat.

6.4 Metodologia i rigor

De cara a un correcte i fluid desenvolupament del projecte, cal una metodologia de treball que s'adapti al problema en qüestió i permeti minimitzar la fricció entre els diferents aspectes del desenvolupament. En aquest apartat es descriu la metodologia de treball utilitzada així com les eines de seguiment i la metodologia de validació.

6.4.1 Metodologia de treball

Tot i que els requisits del sistema a dissenyar, a grans trets, són relativament estàtics: "Dissenyar un algorisme que resolgui un problema concret", quan mirem les tasques a realitzar amb un nivell de granularitat més fina veiem que els requisits del projecte poden variar notablement. Les diferents eines que s'han utilitzat, i decidir com s'ha modelat el problema en els diferents punts de l'algorisme han produït canvis en la planificació per diversos motius: Ja sigui perquè s'ha trobat una idea millor, o perquè han aparegut nous angles al problema que fan que decisions que s'havien pres inicialment no encaixin del tot amb els nous requisits. És per això que aquest projecte s'ha desenvolupat seguint una metodologia de desenvolupament àgil [13].

El projecte s'ha desenvolupat amb iteracions d'una o dues setmanes. Al principi de cada iteració es planteja la llista de tasques a realitzar i al final es realitza una avaluació de la feina feta, replanificant adientment. La planificació i seguiment de les tasques es realitza utilitzant una taula *Kanban*, que permet fer una planificació de tasques concretes a curt termini, acompanyada de la planificació general per tenir una visió de l'estat global. A més, es realitza un control estricte del temps durant les estones de treball inserint descansos periòdics per millorar la productivitat[10].

6.4.2 Eines de seguiment

Cada setmana es fa una reunió de seguiment amb els directors del projecte, coincidint amb l'inici d'una nova iteració de desenvolupament. En les reunions, es mostren els resultats obtinguts, l'estat del projecte, i es plantegen tots els problemes o dubtes que han pogut aparèixer. A continuació, es fa una valoració de la feina realitzada respecte als objectius marcats a la setmana anterior. Finalment, es decideix la llista d'objectius a assolir per la setmana vinent: Funcionalitats del programa, possibles experiments, i/o documentació.

[10] Aquesta tècnica és coneguda com a *timeboxing*.

Tret de les reunions setmanals, la resta de comunicació s'ha fet per correu electrònic. L'objectiu és plantejar els temes importants com més aviat millor de cara a rebre *feedback* ràpidament, i si escau, resoldre els problemes o dubtes abans de la següent reunió amb l'objectiu de fer avançar el projecte de manera fluida.

Finalment, donat que aquest projecte forma part d'un projecte més gran de diferents alumnes treballant en l'àmbit de NLP per a BPM, amb menys freqüència, s'han fet reunions amb tots els membres amb l'objectiu d'agafar perspectiva en els projectes individuals de cadascú i per veure si hi ha alguna part comuna que es pugui compartir.

6.4.3 Mètode de validació

En el cas d'aquest projecte, la validació de resultats és una tasca complicada. Per analitzar la qualitat dels resultats obtinguts de manera objectiva i automàtica caldria una eina que resolgui el problema que es planteja resoldre aquest projecte. És per això que serà necessaria una validació manual.

Així doncs, per avaluar la qualitat de les solucions s'usarà un conjunt suficientment gran d'exemples diversos, alternant casos amb i sense inconsistències. Aquest conjunt està disponible a l'annex A.

A partir d'aquests exemples, s'han plantejat els diversos experiments del capítol 5, juntament amb una validació prèvia dels resultats. Com que la similaritat entre un model BPMN i un document escrit és difícil de determinar, en alguns casos ha calgut recorrer a un expert[11] que corrobori els resultats del software desenvolupat.

6.5 Integració de coneixements

Aquest projecte és una combinació de dues disciplines concretes de la informàtica. El *Natural Language Processing* i el *Business Process Management*. Els dos camps es troben en recerca activa i els resultats obtinguts en aquests utilitzen tot tipus de tècniques de diferents camps com la intel·ligència artificial, la teoria de grafs o l'algorísmia.

Per a l'elaboració del projecte s'utilitzen, o s'han plantejat utilitzar[12] les següents tècniques i algorismes relacionades amb els continguts de la carrera:

- Freeling, com a analitzador semàntic.
- Solver de Constraint Programming.
- Mètriques de distancia.
- Vectors de característiques.
- Similaritat semàntica en Wordnet.
- Càlcul de Behavioral Profiles en grafs de processos.
- Algorismes de grafs de flux, concretament el d'eliminació de back-edges.
- Linear Assignment Problem i l'Hungarian Method.
- Programació funcional.

[11] Els experts han estat en aquest cas els directors del projecte.

[12] No totes les alternatives plantejades han acabat al projecte original, però això no vol dir que no s'hagin estudiat i valorat.

6.6 Lleis i regulacions

Pel que fa al software desenvolupat, aquest és una eina que està planejada per assistir en tècniques de Business Process Management. Actualment no existeix cap legislació que reguli com una empresa ha de documentar els seus processos des d'un punt de vista de BPM, i si es requerís d'alguna documentació en un camp concret aquesta existeix separada de la documentació formal dels processos de negoci. Així doncs podem dir que no hi ha cap normativa que s'hagi de tenir en compte a l'hora de parlar del programa que s'ha desenvolupat.

No obstant, pel que fa al desenvolupament del software i les opcions de comercialització, cal tenir en compte quines són les llicències utilitzades en totes les llibreries que aquest integra, i quina serà la llicència del producte final. S'ha d'anar en compte amb no incumplir cap dels termes i condicions de totes les llicències que s'estàn fent servir.

Capítol 7

Conclusions

L'objectiu d'aquest projecte ha estat desenvolupar un algorisme per comparar descripcions textuals informals i models BPMN formals. Podem concloure que l'objectiu principal s'ha assolit satisfactòriament. L'algorisme desenvolupat és capaç d'agrupar parelles heterogènies de models i textos (secció 5.1) i es pot utilitzar com una eina per assistir a la detecció d'inconsistències entre els dos tipus de documentació.

Tornant als objectius específics del treball (secció 2.3), podem dir que s'han assolit amb un alt nivell de satisfacció. Alguns dels objectius addicionals de l'apartat 2.4 es plantegen com a treball futur, mentre que la resta s'han dut a terme dins el període establert.

La utilització de tècniques de NLP és un dels aspectes principals del projecte. Tot el text, tant del model com del document, s'ha analitzat amb *Freeling*. S'ha fet ús extensiu dels textos analitzats a la fase d'extracció de característiques. D'altra banda, el tractament de models BPMN també ha estat un punt molt important en la implementació final, incorporant tècniques de tractament de grafs i l'algorisme dels *Behavioral Profiles* (secció 3.5.2). Es pot dir que aquest projecte integra els camps *Business Process Management* i el *Natural Language Processing* en un mateix sistema de manera satisfactòria.

Tal com s'havia plantejat, l'enfoc de l'algorisme es basa en crear una representació homogènia amb vectors de característiques que redueixin a la mateixa representació el model BPMN i la descripció textual. Aquest enfoc ha permès facilitar substancialment la tasca de trobar una puntuació de similitud.

Respecte establir un ordre entre el text i el model podem dir que, per al cas del model, aquest objectiu s'ha assolit amb un bon nivell de qualitat. Es calcula un ordre parcial bàsic amb el *Behavioral Profile* i aquest ordre s'amplia amb la informació addicional dels *Message Flows* utilitzant un algorisme personalitzat. Pel que fa a l'ordenació de frases, s'ha optat per un enfoc bàsic però el resultat final és efectiu.

L'algorisme d'optimització per trobar la correspondència entre tasques i frases ha estat un dels punts forts del projecte i on s'hi ha invertit més temps d'experimentació. Durant el desenvolupament s'han plantejat diversos tipus d'alternatives, de les quals LAP, CSP i ILP han estat les principals (secció 4.6). Utilitzant ILP, s'ha trobat una implementació que compleix les restriccions definides a l'apartat 3.6 i que a més ho fes amb un nivell d'eficiència molt superior a l'esperat. Per aquest fet, podem dir que el grau de satisfacció d'aquesta part és molt elevat. A més, s'han implementat dues altres

alternatives també viables. En cas que creixés considerablement la mida del problema es pot utilitzar LAP a canvi d'una pèrdua d'optimalitat. D'altra banda, si cal introduir noves restriccions més complexes al problema, CP quedaria com una solució alternativa menys eficient però molt més expressiva.

Finalment, la detecció d'inconsistències s'ha realitzat de manera implícita. En comptes de retornar una llista d'errors a l'usuari, l'algorisme mostra l'assignació òptima calculada a l'usuari, oferint informació detallada de perquè ha assignat cadascuna de les tasques. Aquesta informació es pot utilitzar per assistir a l'usuari a l'hora de detectar problemes entre el model BPMN i la descripció textual (veure secció 5.3).

Com a possibles millores a l'algorisme desenvolupat, possiblement la major contribució immediata a millorar la qualitat de l'algorisme és adreçar els problemes que sorgeixen durant la discussió a l'experiment de la secció 5.2. Pot ser també molt interessant experimentar amb noves mètriques de similitud i tipus de característiques. Finalment, una possible ampliació futura seria integrar un mètode més sofisticat per a l'ordenació temporal de frases en un text en llenguatge natural.

El tractament de processos de negoci utilitzant tècniques d'*NLP* és un camp emergent. L'eina desenvolupada en aquest projecte, que per mesurar la similitud entre una descripció textual i un model BPMN, pot resultar molt útil a l'hora de validar els resultats de diversos algorismes, com el de traducció automàtica de text a model BPMN. Per altra banda, aquesta eina es pot utilitzar per ajudar a detectar inconsistències de manera automàtica entre diferents tipus de documentació en una empresa real. Aquest projecte, com la resta d'estudis en aquest camp, té per objectiu la creació d'eines que assisteixin a la presa de decisions i ajudin a optimitzar els processos de negoci. Aquest enfoc algorísmic al problema pot obrir la porta a millores considerables en el funcionament de les institucions i empreses.

Bibliografia

[1] Han Aa, Henrik Leopold i Hajo A. Reijers. "Business Process Management: 13th International Conference, BPM 2015, Innsbruck, Austria, August 31 - September 3, 2015, Proceedings". A: Cham: Springer International Publishing, 2015. Cap. Detecting Inconsistencies Between Process Models and Textual Descriptions, pàg. 90 - 105. ISBN: 9783319230634. DOI: 10.1007/978-3-319-23063-4_6. URL: http://dx.doi.org/10.1007/978-3-319-23063-4_6.

[2] Han van der Aalst, Henrik Leopold i Hajo A. Reijers. "Dealing with Behavioral Ambiguity in Textual Process Descriptions". A: ().

[3] Wil MP van der Aalst et al. "Soundness of workflow nets: classification, decidability, and analysis". A: *Formal Aspects of Computing* 23.3 (2011), pàg. 333 - 363.

[4] A.V. Aho, R. Sethi i J.D Ullman. *Compilers: Principles, techniques and tools*. 2a ed. The address: Addison-Wesley, 2007. ISBN: 9780321491695.

[5] *BPMN for research*. 2016. URL: https://github.com/camunda/bpmn-for-research.

[6] Cristina Cabanillas et al. "RALph: A Graphical Notation for Resource Assignments in Business Processes". A: *Advanced Information Systems Engineering - 27th International Conference, CAiSE 2015, Stockholm, Sweden, June 8-12, 2015, Proceedings*. 2015, pàg. 53 - 68. DOI: 10.1007/978-3-319-19069-3_4. URL: http://dx.doi.org/10.1007/978-3-319-19069-3_4.

[7] Michael Collins. *Machine Learning Methods in Natural Language Processing*.

[8] R. Dechter. "Constraint Processing (The Morgan Kaufmann Series in Artificial Intelligence)". A: (mai. de 2003).

[9] Fabian Fredrich. *Text2Process Test Data Models*. URL: https://github.com/FabianFriedrich/Text2Process/.

[10] Fabian Friedrich, Jan Mendling i Frank Puhlmann. "Process model generation from natural language text". A: *Advanced Information Systems Engineering*. Springer. 2011, pàg. 482 - 496.

[11] Felix Kossak et al. *A Rigorous Semantics for BPMN 2.0 Process Diagrams*. 2014. ISBN: 9783319099309.

[12] H. Leopold, J. Mendling i A. Polyvyanny. "Supporting process model validation through natural language generation". A: (2014).

[13] *Manifesto for Agile Software Developement*. http://www.agilemanifesto.org/. Accedit a: 28-02-2016.

[14] Cristopher D. Manning, Mihai Surdeanu i John Bauer. "The Stanford CoreNLP Natural Language Processing Toolkit". A: (2014).

[15] *Mining similarity using euclidean distance, pearson correlation and filtering.* 2010. URL: http://mines.humanoriented.com/classes/2010/fall/csci568/portfolio_exports/mvoget/similarity/similarity.html.

[16] James Munkres. "Algorithms for the Assignment and Transportation Problems". A: *Journal of the Society for Industrial and Applied Mathematics* 5.1 (1957), pàg. 32 - 38. ISSN: 03684245. URL: http://www.jstor.org/stable/2098689.

[17] Lluís Padró i Evgeny Stanilovsky. "FreeLing 3.0: Towards Wider Multilinguality. Proceedings of the Language Resources and Evaluation Conference (LREC 2012) ELRA." A: (2012).

[18] Lluís Padró i Jordi Turmo. "TextServer: Cloud-based Multilingual Natural Language Processing. Proceedings of the 15th IEEE International Conference on Data Mining (ICDM'15) IEEE." A: (2012).

[19] Alexander Schrijver. *Theory of linear and integer programming.* John Wiley & Sons, 1998.

[20] Sergey Smirnov, Matthias Weidlich i Jan Mendling. "Business process model abstraction based on behavioral profiles". A: *Service-Oriented Computing.* Springer, 2010, pàg. 1 - 16.

[21] Richard Socher. *Deep Learning for NLP without magic.* 2014.

[22] Roman V. Yampolskiy. "Turing Test as a Defining Feature of AI-Completeness". A: (2012).

Apèndix A

Parelles Model-Text utilitzades als experiments

En aquest apèndix es recullen les difents parelles model-text utilitzades en els experiments de la secció 5. Els quatre primers provenen de la col·lecció "BPMN For Research" de Camunda [5]. Els quatre següents s'han convertit manualment a bpmn a partir del conjunt de dades d'exemple de [9] utilitzant el software de modelat de BPMN de Camunda, *bpmn.io*. L'exemple: "Hospital", s'ha obtingut de l'article disponible a [6]. Pel que fa a l'última parella model-text, "Zoo", aquesta s'ha modelat manualment a partir de l'explicació d'un cas real de procés de negoci al Zoo de Barcelona[1].

Per facilitar la lectura d'aquest document en format imprès s'inclouen els diagrames BPMN ampliats en fulls apaisats al final d'aquest annex.

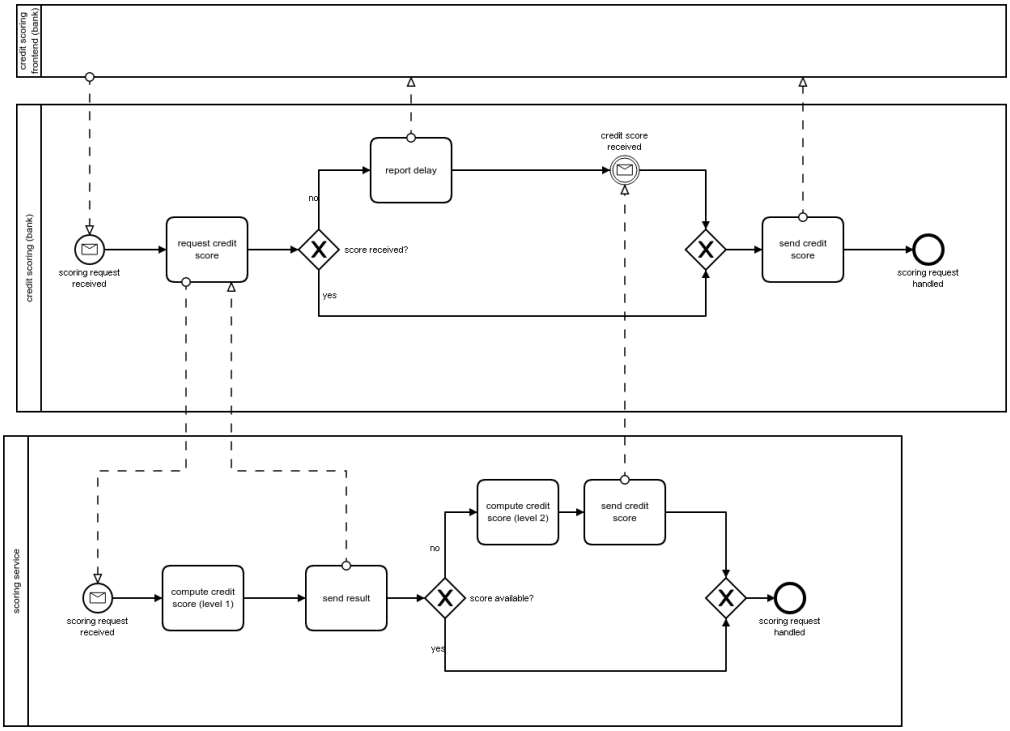
[1]Les fonts de les quals s'ha obtingut la descripció del procés de negoci no són oficials. Per aquest fet, aquest procés es considerarà com un exemple acadèmic.

A.1 Credit Scoring [CRED]

Descripció textual

The sales clerks in a bank can use their software frontend to receive the credit-scoring for a certain customer. This starts a process in the banking system which communicates with the agency in the background. This process sends a scoring request to the agency right after the beginning. Then, the Agency does a first quick scoring (level 1). This will often lead to an immediate result which is then returned directly to the banking system within seconds. The banking process presents the result to the clerk sitting at the frontend. Sometimes the scoring cannot be determined immediately and takes longer. In this case the agency informs the banking process of the delay and then starts the level 2 scoring (which can take up to a couple of minutes). After the scoring result is determined, the information is sent back to the banking process. The banking process displays a message to the clerk when he receives information about the delay to check again later. As soon as the result arrives, it can be seen at the frontend.

Model BPMN

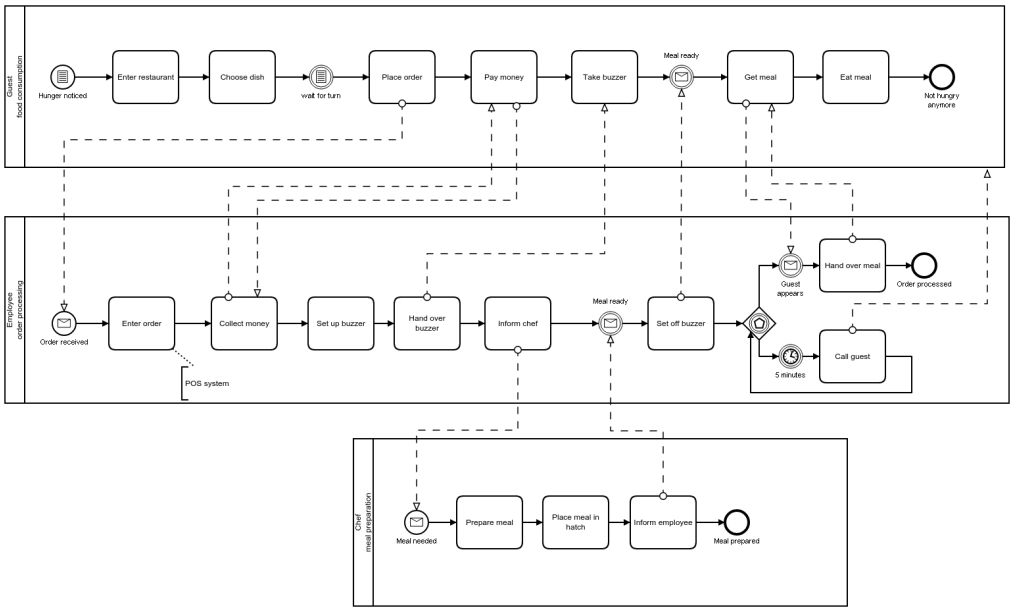


A.2 Self Service Restaurant [REST]

Descripció textual

A guest enters the restaurant when feeling hungry. He chooses a dish from the changing meal range and waits until it is his turn. Following this he places his order with the employee. The employee enters the order into the POS system and collects the money from the guest. After the payment, the employee sets up a buzzer and passes it on to the guest with the following information: "When the buzzer rings, your dinner is ready". Afterwards the employee informs the chef of the new meal order. The chef prepares the meal and places it in the service hatch. He then informs the employee that he has placed the finished meal in the service hatch. As soon as the employee is aware that the meal is ready he sets off the guest's buzzer. This is how the guest finds out that his meal is ready for collection. He can pick up his meal and eat it. As soon as the guest appears at the service hatch, the employee hands over his meal. Should a guest not react to the buzzer, the employee calls for him after 5 minutes, if necessary several times in a row.

Model BPMN

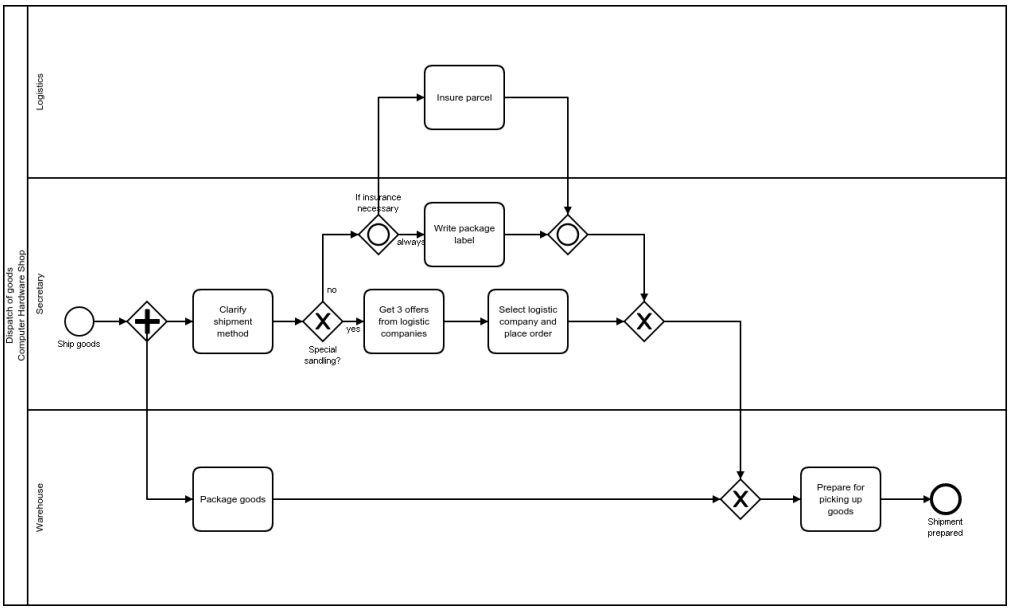


A.3 Dispatch of Goods [DISP]

Descripció textual

If goods shall be shipped, the secretary clarifies who will do the shipping. If you have large amounts, special shipping will be necessary. In these cases the secretary invites three logistic companies to make offers and she selects one of them. In case of small amounts, normal post shipment is used. Therefore a package label is written by the secretary and a parcel insurance taken by the logistics department head if necessary. In the meantime the goods can be already packaged by the warehousemen. If everything is ready, the packaged goods are prepared for being picked up by the logistic company.

Model BPMN

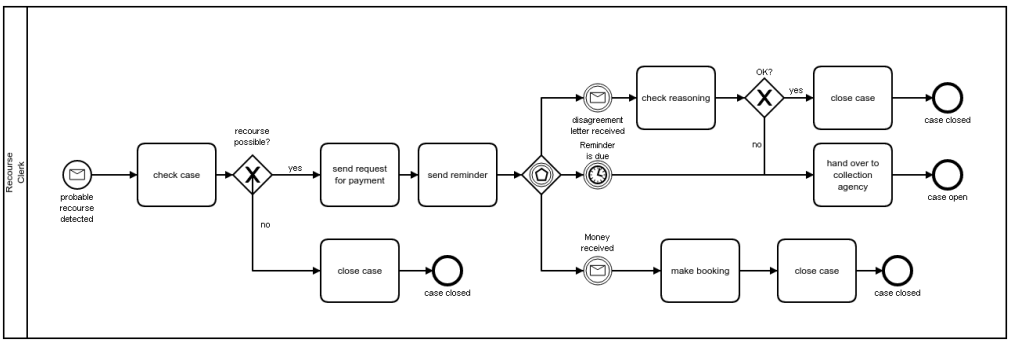


A.4 Recourse [RECO]

Descripció textual

If an insurant could be possibly subrogated against, I get information about that. I check that case and if the possibility is really there, I send a request for payment to the insurant and make me a reminder. If recourse is not possible, I close the case. When we receive the money, I make a booking and close the case. If the insurant disagrees with the recourse, I'll have to check the reasoning of that. If he is right, I simply close the case. If he is wrong, I forward the case to a collection agency. It the deadline for disagreement is reached and we haven't received any money, I forward the case to the collection agency as well.

Model BPMN

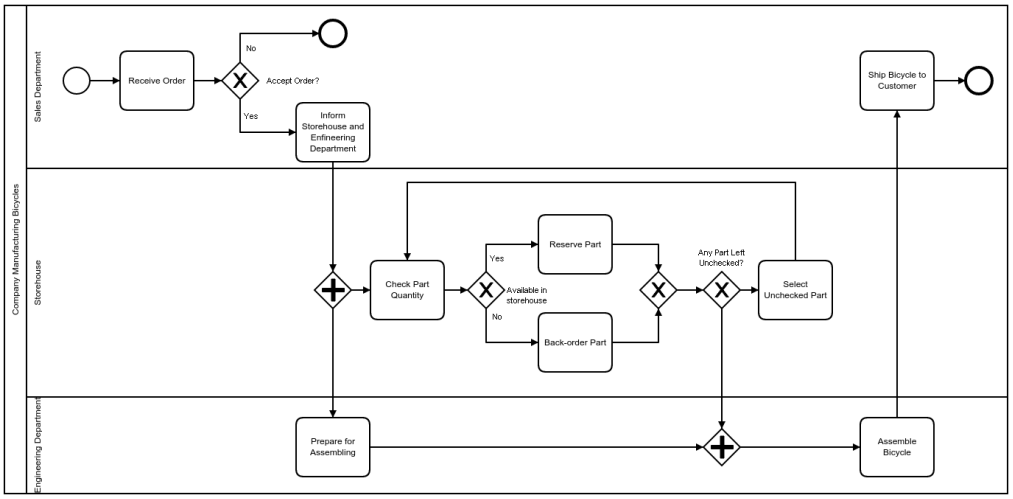


A.5 Bicycle Manufacturer [BICL]

Descripció textual

A small company manufactures customized bicycles. Whenever the sales department receives an order, a new process instance is created. A member of the sales department can then reject or accept the order for a customized bike. In the former case, the process instance is finished. In the latter case, the storehouse and the engineering department are informed. The storehouse immediately processes the part list of the order and checks the required quantity of each part. If the part is available in-house, it is reserved. If it is not available, it is back-ordered. This procedure is repeated for each item on the part list. In the meantime, the engineering department prepares everything for the assembling of the ordered bicycle. If the storehouse has successfully reserved or back-ordered every item of the part list and the preparation activity has finished, the engineering department assembles the bicycle. Afterwards, the sales department ships the bicycle to the customer and finishes the process instance.

Model BPMN

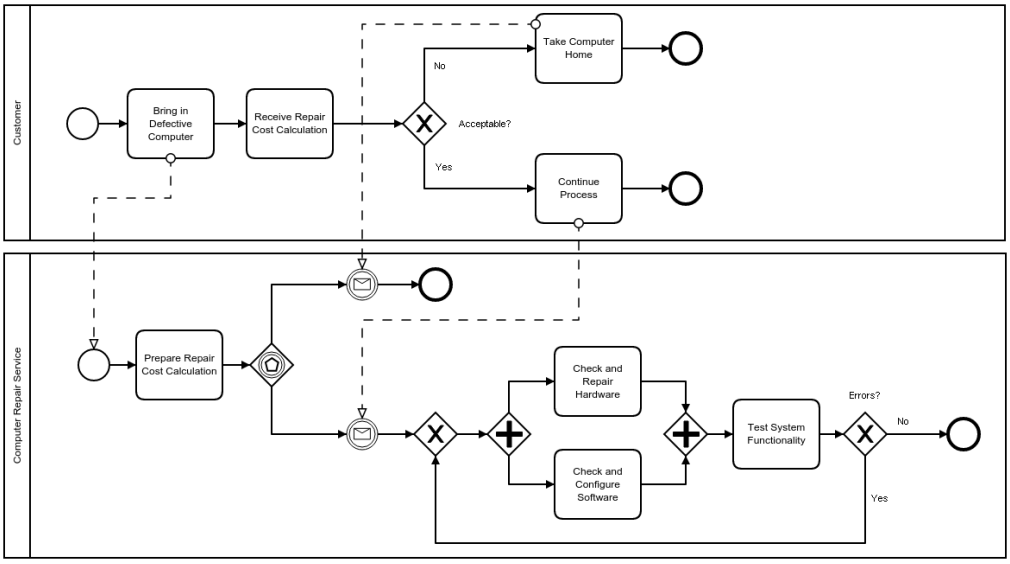


A.6 Computer Repair Shop [COMP]

Descripció textual

The workflow of a computer repair service (CRS) can be described as follows. A customer brings in a defective computer and the CRS checks the defect and hands out a repair cost calculation back. If the customer decides that the costs are acceptable, the process continues, otherwise she takes her computer home unrepaired. The ongoing repair consists of two activities, which are executed, in an arbitrary order. The first activity is to check and repair the hardware, whereas the second activity checks and configures the software. After each of these activities, the proper system functionality is tested. If an error is detected another arbitrary repair activity is executed, otherwise the repair is finished.

Model BPMN

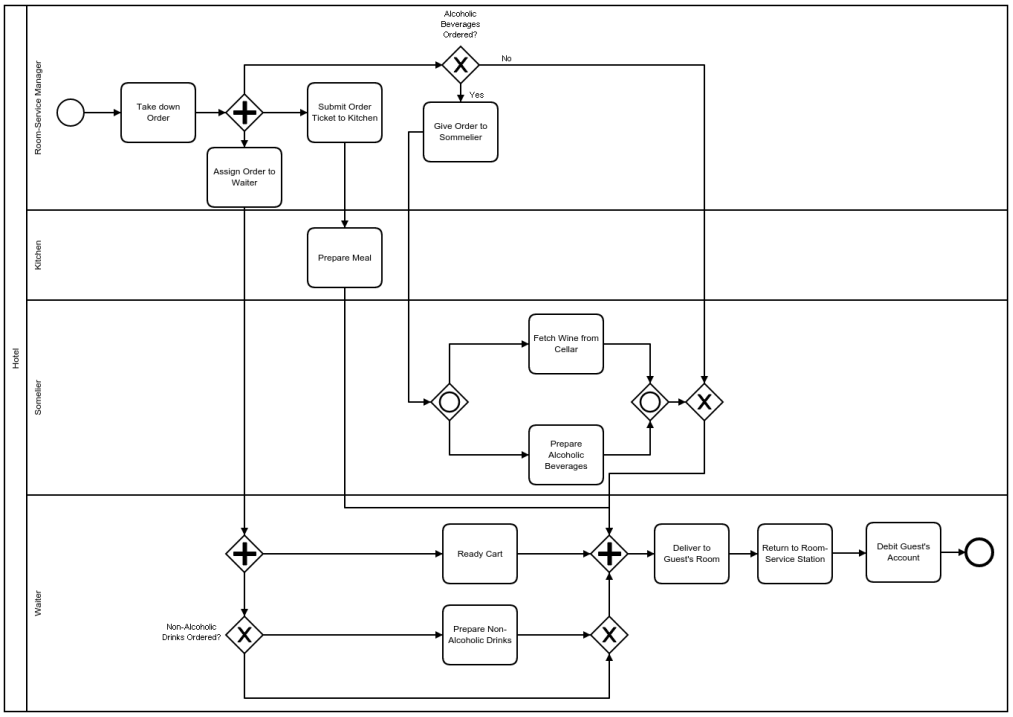


A.7 Hotel [HOTL]

Descripció textual

The Evanstonian is an upscale independent hotel. When a guest calls room service at The Evanstonian, the room-service manager takes down the order. She then submits an order ticket to the kitchen to begin preparing the food. She also gives an order to the sommelier (i.e., the wine waiter) to fetch wine from the cellar and to prepare any other alcoholic beverages. Eighty percent of room-service orders include wine or some other alcoholic beverage. Finally, she assigns the order to the waiter. While the kitchen and the sommelier are doing their tasks, the waiter readies a cart (i.e., puts a tablecloth on the cart and gathers silverware). The waiter is also responsible for nonalcoholic drinks. Once the food, wine, and cart are ready, the waiter delivers it to the guest's room. After returning to the room-service station, the waiter debits the guest's account. The waiter may wait to do the billing if he has another order to prepare or deliver.

Model BPMN



A.8 Underwriter [UNWR]

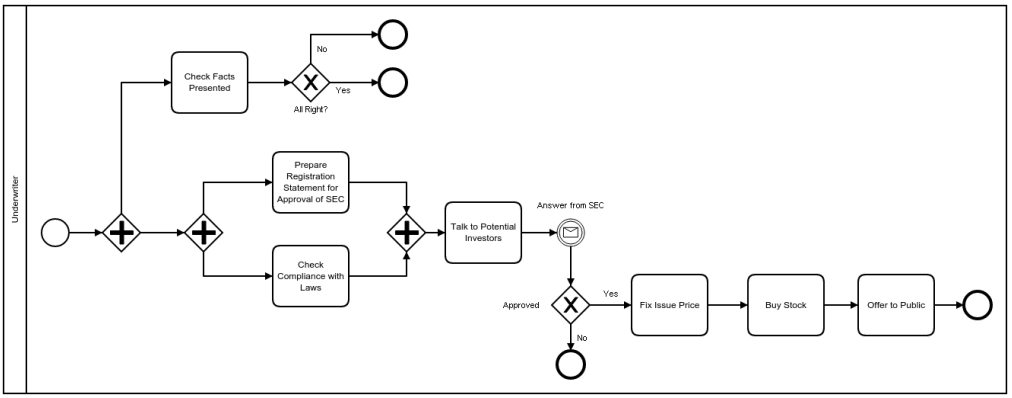
Descripció textual

Whenever a company makes the decision to go public, its first task is to select the underwriters. Underwriters act as financial midwives to a new issue. Usually they play a triple role: First they provide the company with procedural and financial advice, then they buy the issue, and finally they resell it to the public.

Established underwriters are careful of their reputation and will not handle a new issue unless they believe the facts have been presented fairly. Thus, in addition to handling the sale of a company's issue, the underwriters in effect give their seal of approval to it.

They prepare a registration statement for the approval of the Securities and Exchange Commission (SEC). In addition to registering the issue with the SEC, they need to check that the issue complies with the so-called blue-sky laws of each state that regulate sales of securities within the state. While the registration statement is awaiting approval, underwriters begin to firm up the issue price. They arrange a road show to talk to potential investors. Immediately after they receive clearance from the SEC, underwriters fix the issue price. After that they enter into a firm commitment to buy the stock and then offer it to the public, when they haven't still found any reason not to do it.

Model BPMN

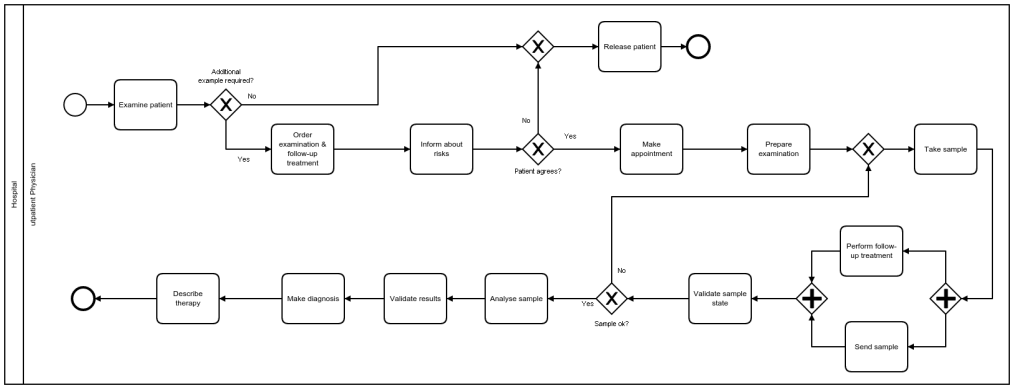


A.9 Hospital [HOSP]

Descripció textual

The examination process can be summarised as follows. The process starts when the female patient is examined by an outpatient physician, who decides whether she is healthy or needs to undertake an additional examination. In the former case, the physician fills out the examination form and the patient can leave. In the latter case, an examination and follow-up treatment order is placed by the physician who additionally fills out a request form. Beyond information about the patient, the request form includes details about the examination requested and refers to a suitable lab. Furthermore, the outpatient physician informs the patient about potential risks. If the patient signs an informed consent and agrees to continue with the procedure, a delegate of the physician arranges an appointment of the patient with one of the wards. The latter is then responsible for taking a sample to be analysed in the lab later. Before the appointment, the required examination and sampling is prepared by a nurse of the ward based on the information provided by the outpatient section. Then, a ward physician takes the sample requested. He further sends it to the lab indicated in the request form and conducts the follow-up treatment of the patient. After receiving the sample, a physician of the lab validates its state and decides whether the sample can be used for analysis or whether it is contaminated and a new sample is required. After the analysis is performed by a medical technical assistant of the lab, a lab physician validates the results. Finally, a physician from the outpatient department makes the diagnosis and prescribes the therapy for the patient.

Model BPMN

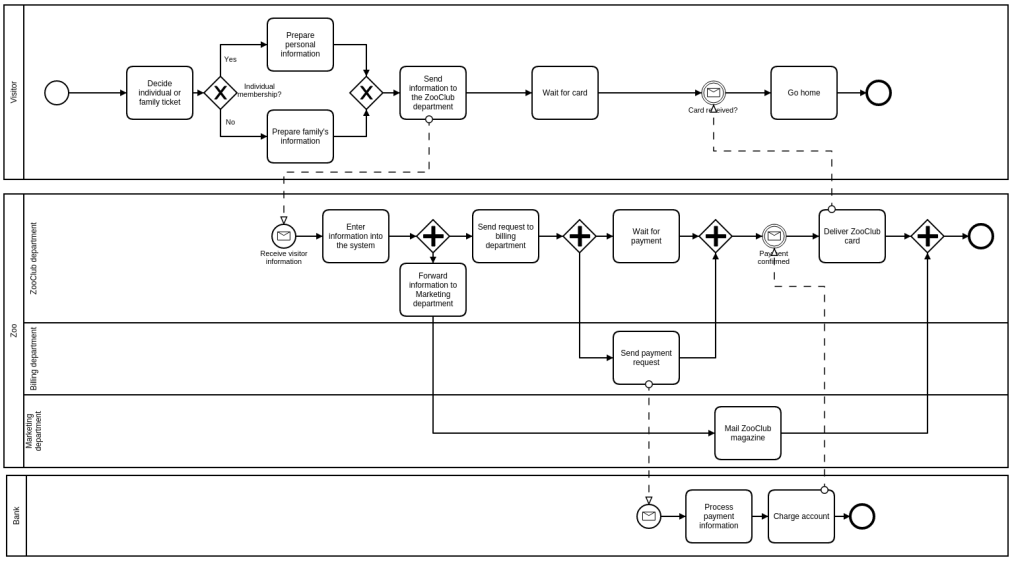


A.10 Zoo [ZOO]

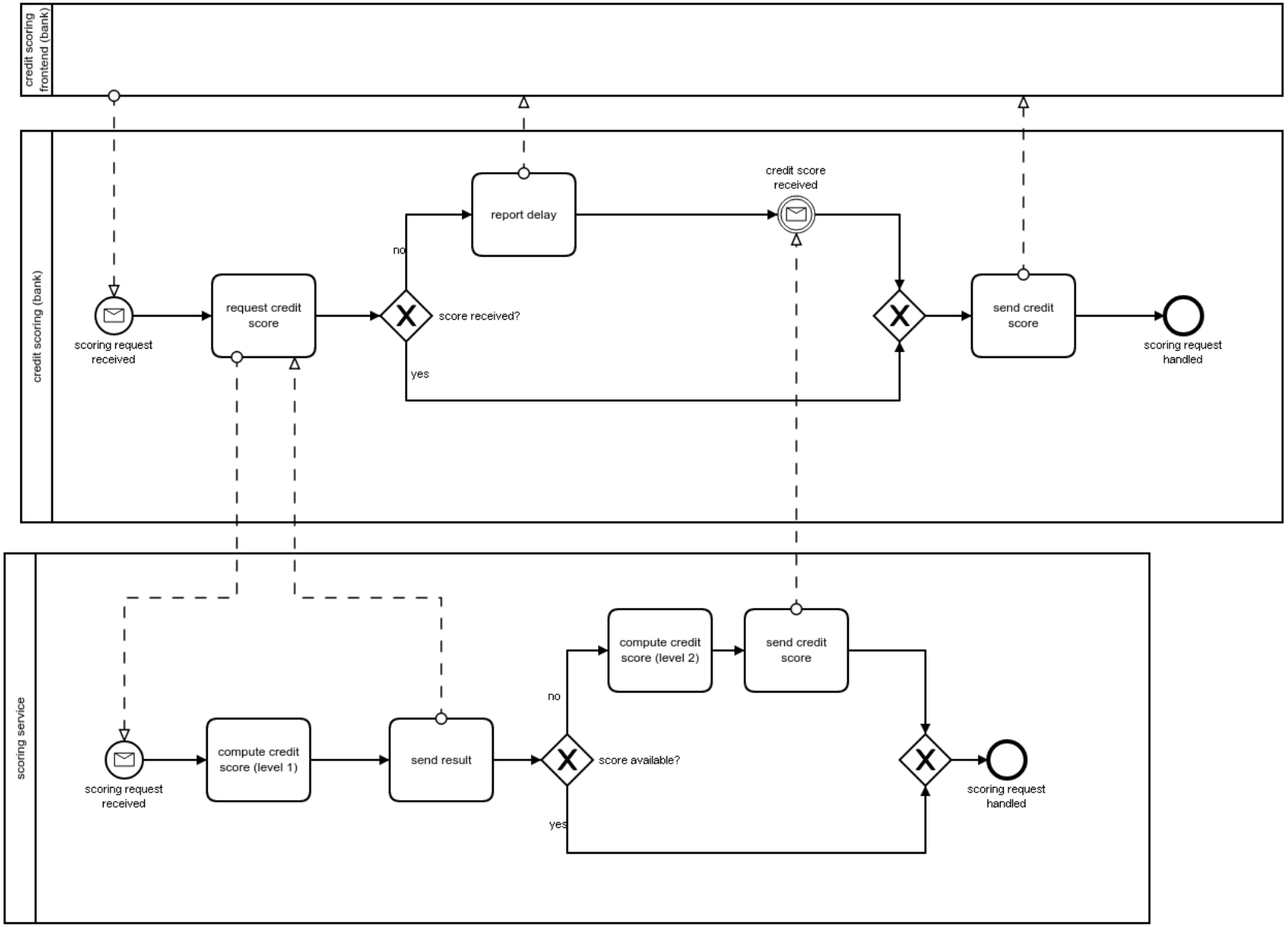
Descripció textual

When a visitor wants to become a member of Barcelona's ZooClub, the following steps must be taken. First of all, the customer must decide whether he wants an individual or family membership. If he wants an individual membership, he must prepare his personal information. If he wants a family membership instead, he should prepare the information for its spouse and spawn as well. The customer must then give this information to the ZooClub department. The ZooClub department introduces the visitor's personal data into the system and takes the payment request to the Billing department. The ZooClub department also forwards the visitor's information to the marketing department. The billing department sends the payment request to the bank. The bank processes the payment information and, if everything is correct, charges the payment into user's account. Once the payment is confirmed, the ZooClub department can print the card and deliver it to the visitor. In the meantime, the Marketing department makes a request to mail the Zoo Club's magazine to the visitor's home. Once the visitor receives the card, he can go home.

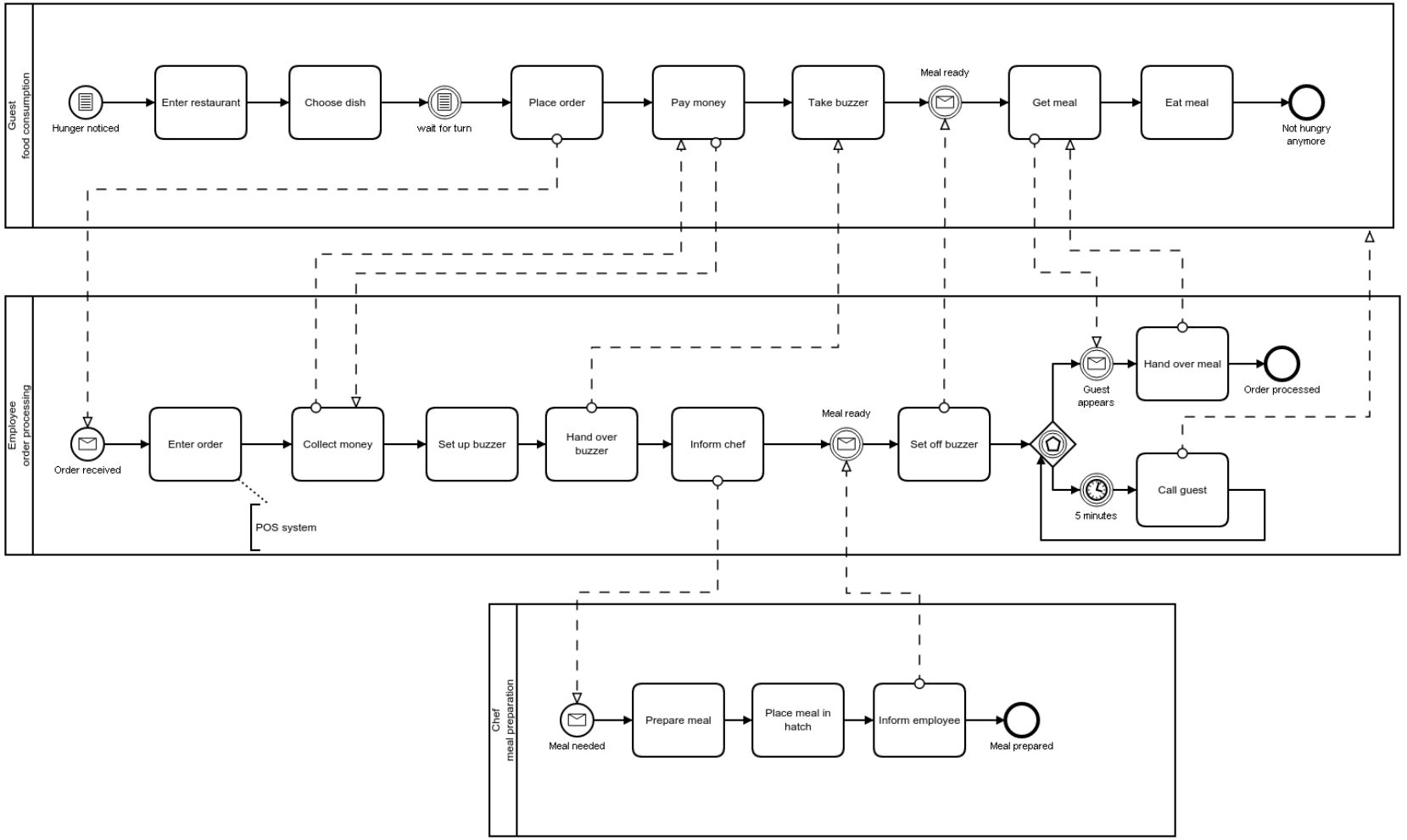
Model BPMN



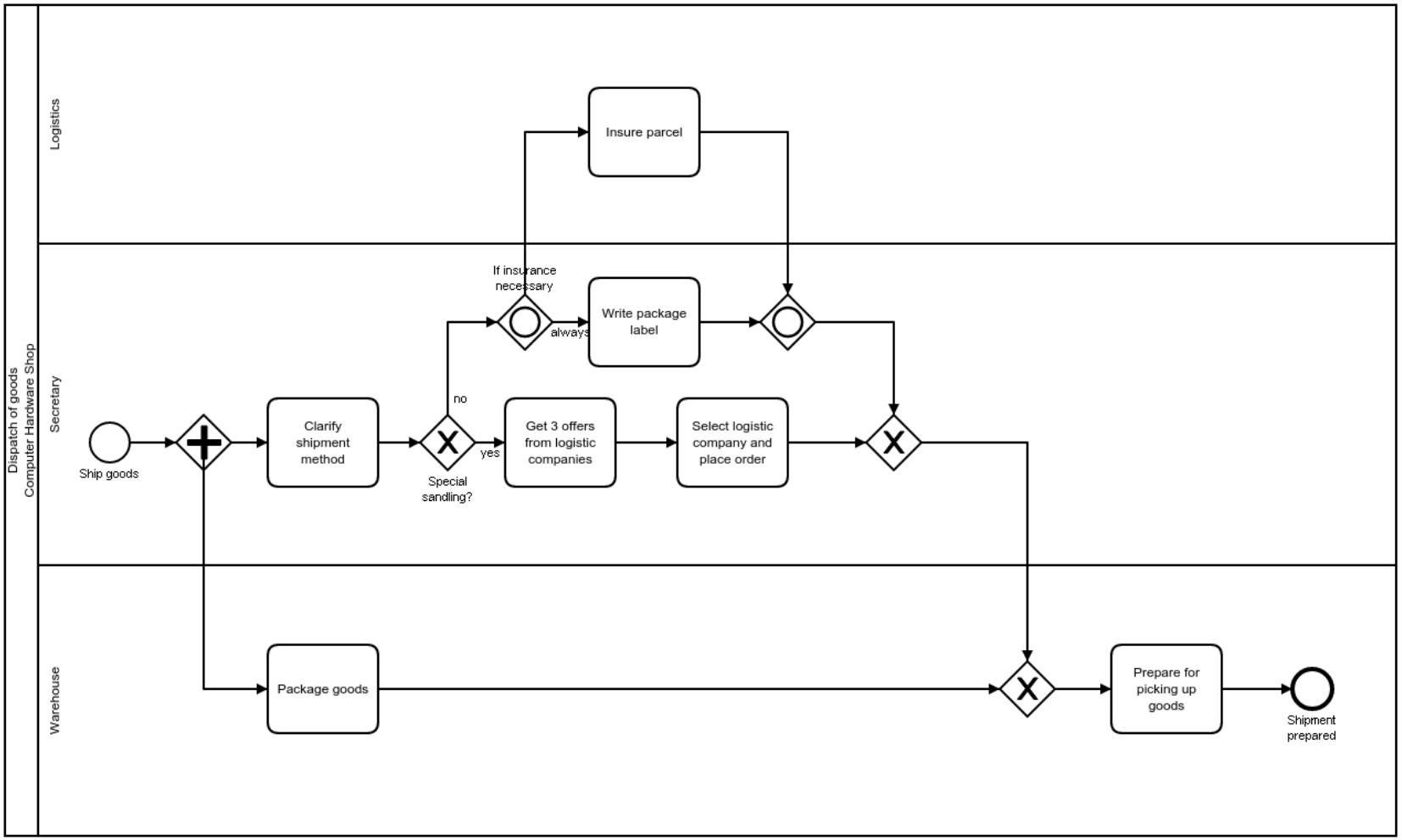
Credit Scoring [CRED]



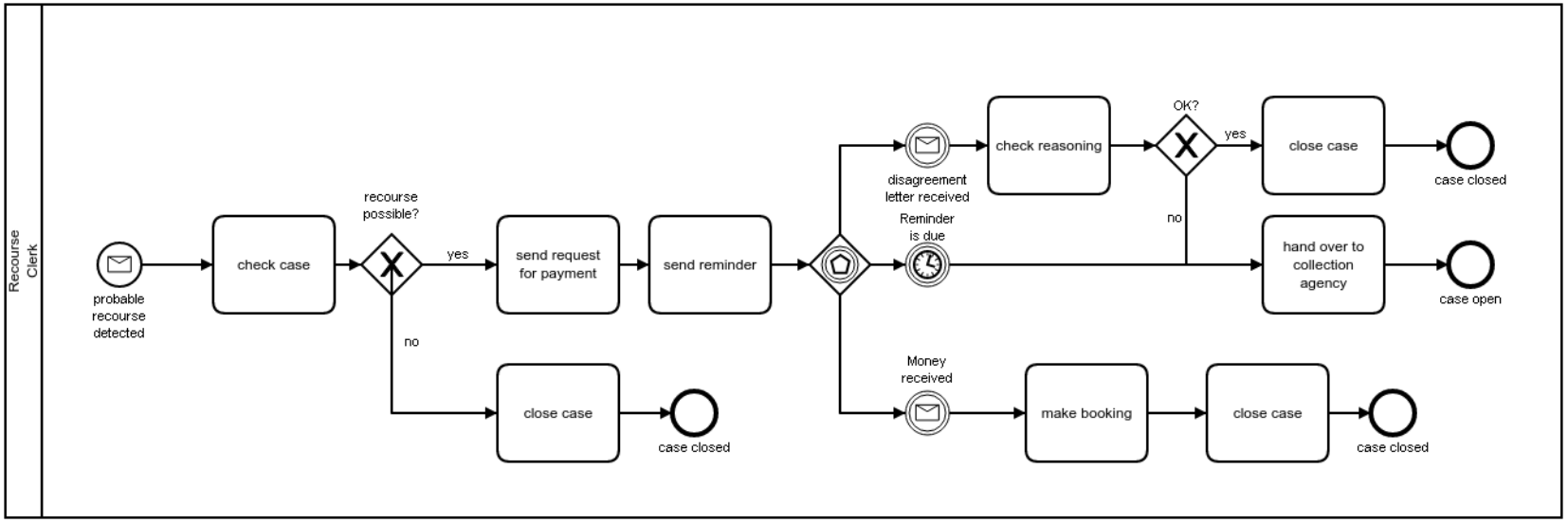
Self Service Restaurant [REST]



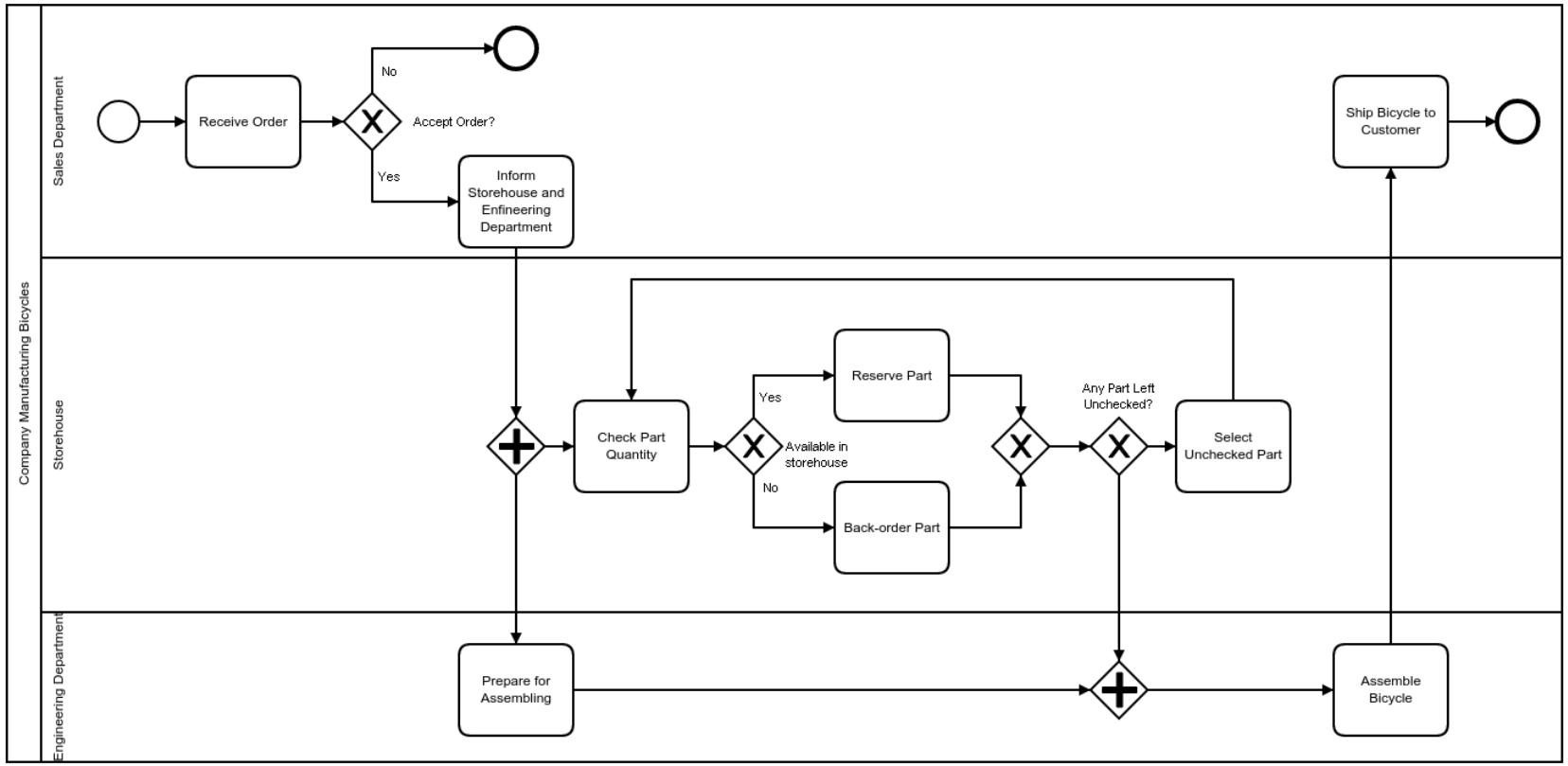
Dispatch of Goods [DISP]



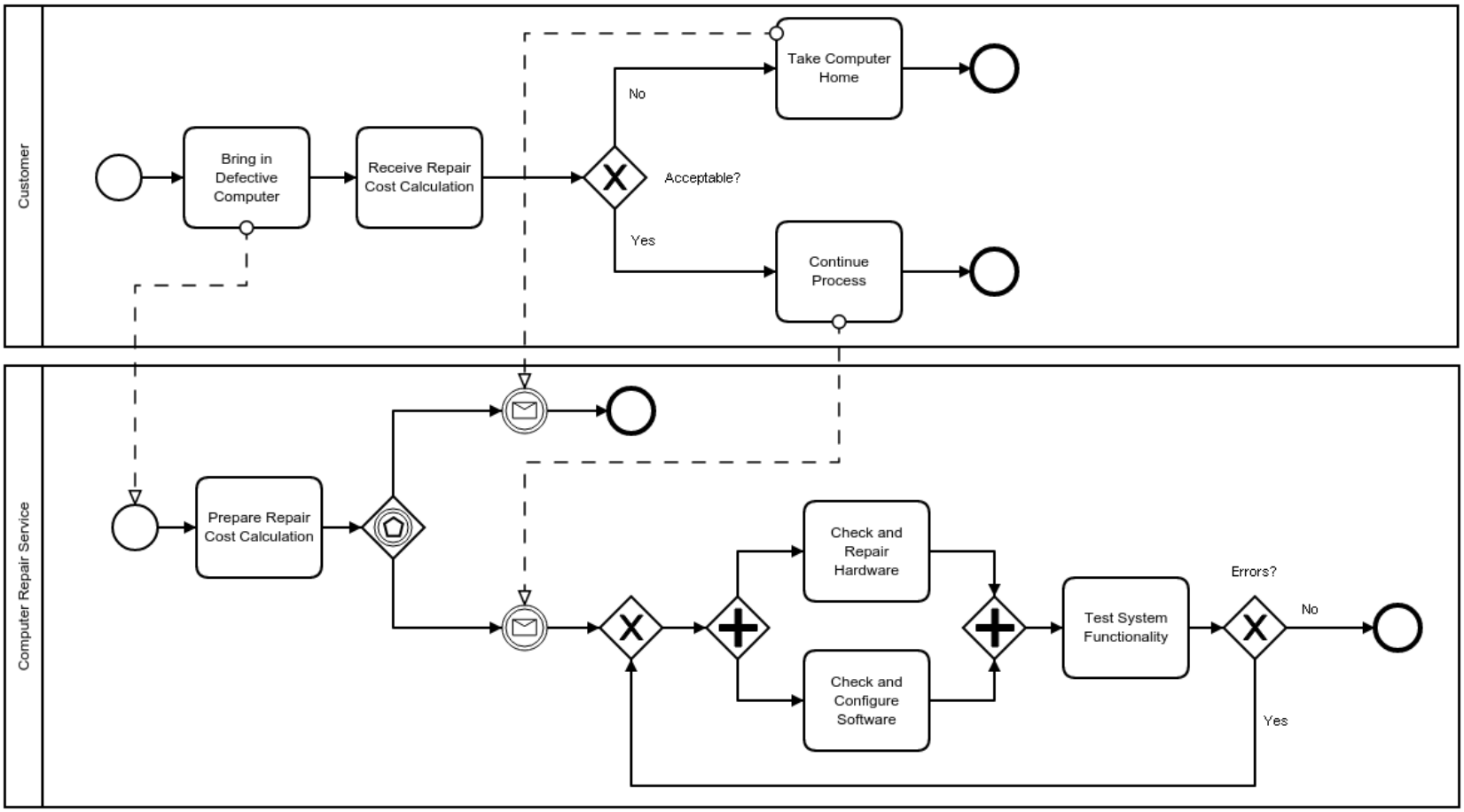
Recourse [RECO]



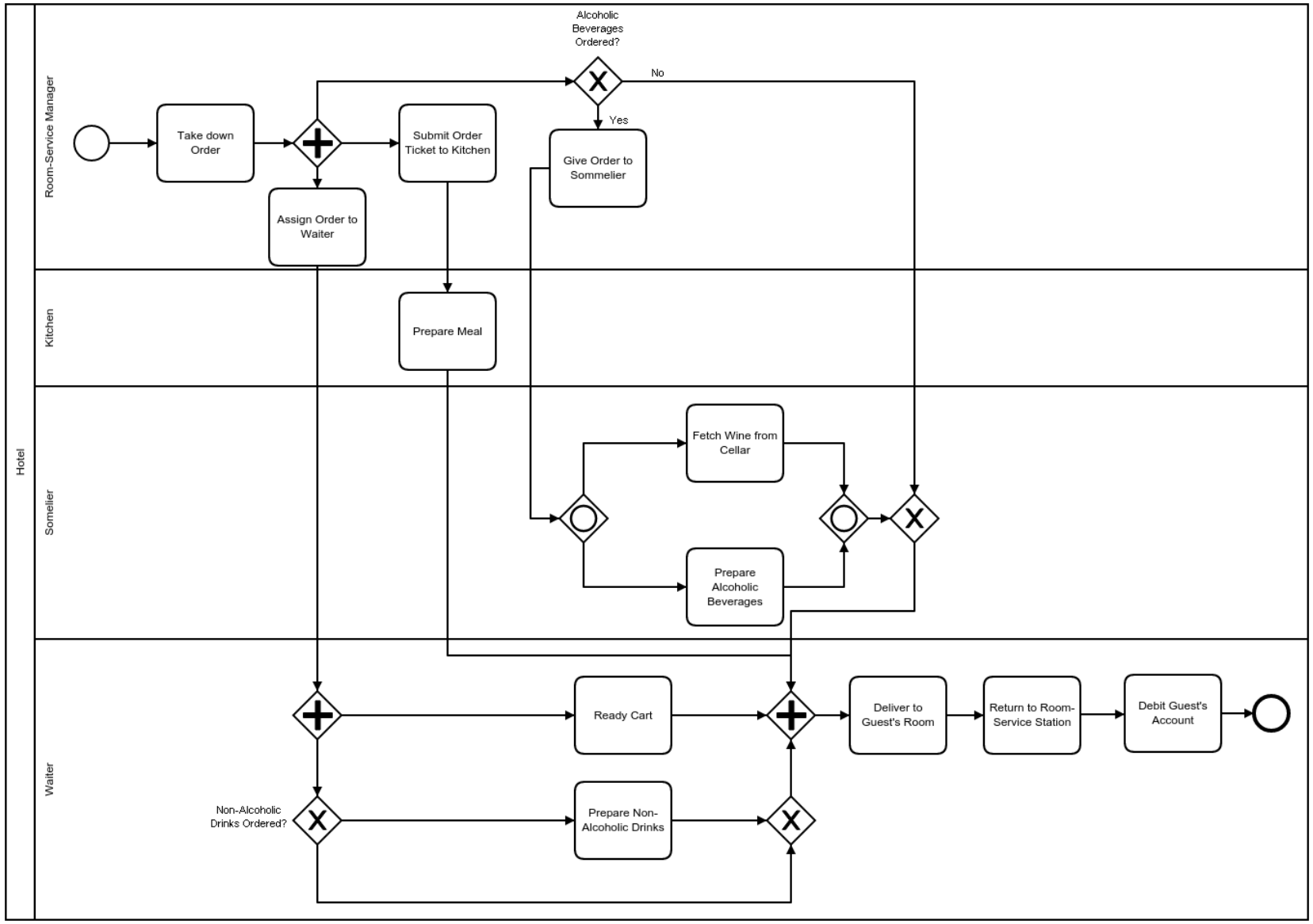
Bicycle Manufacturer [BICL]



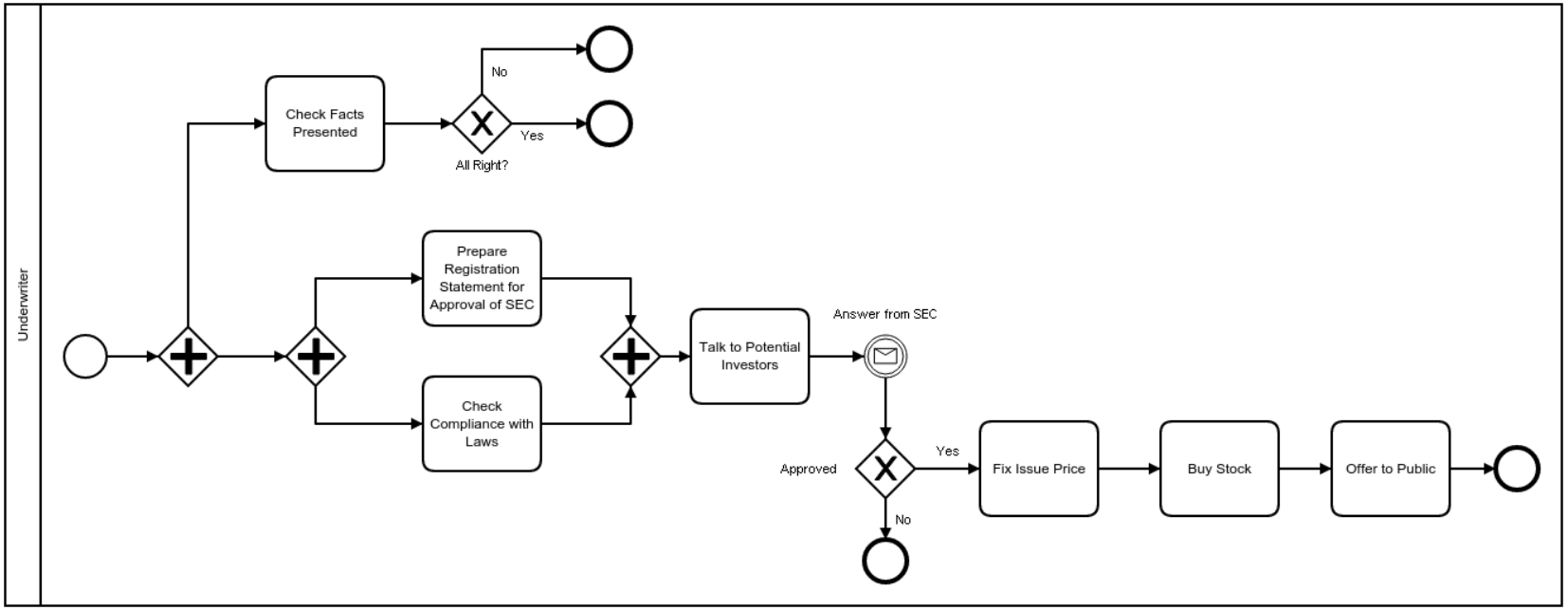
Computer Repair Shop [COMP]



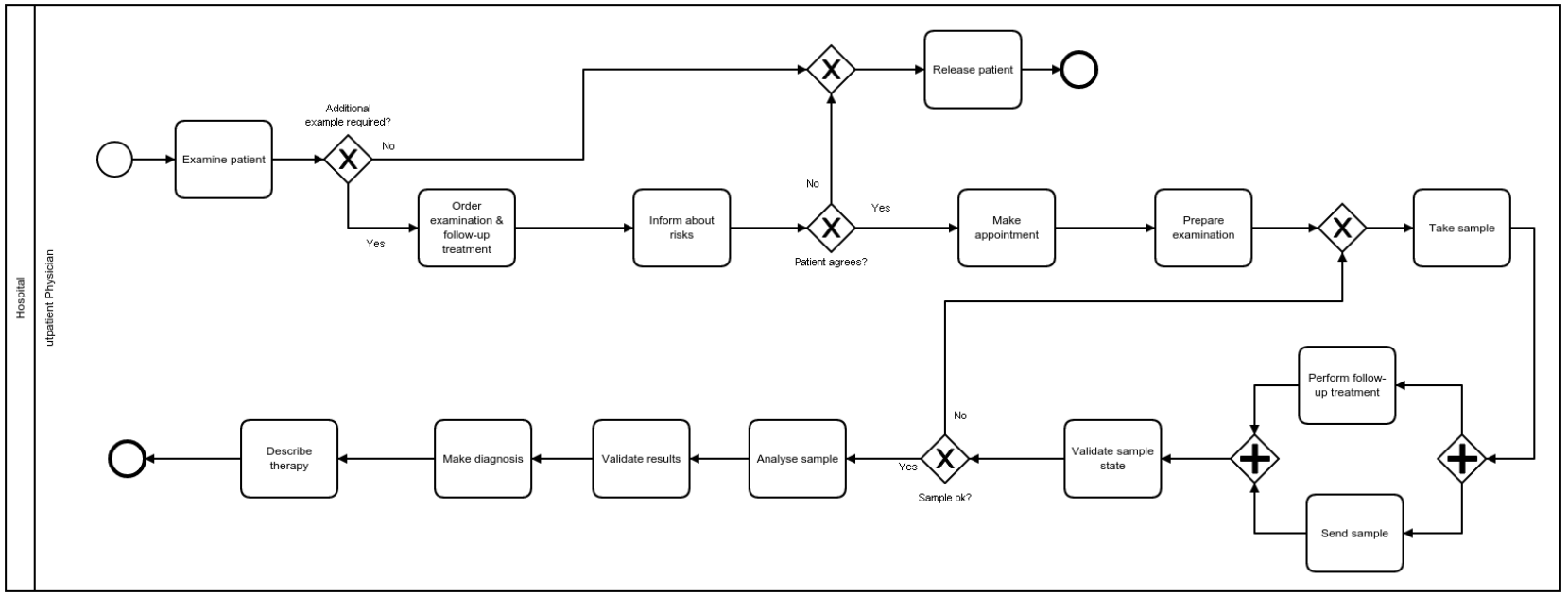
Hotel [HOTL]


Hotel
Room-Service Manager
Kitchen
Somelier
Waiter
Alcoholic Beverages Ordered?
No
Yes
Take down Order
Submit Order Ticket to Kitchen
Assign Order to Waiter
Give Order to Sommelier
Prepare Meal
Fetch Wine from Cellar
Prepare Alcoholic Beverages
Ready Cart
Non-Alcoholic Drinks Ordered?
Prepare Non-Alcoholic Drinks
Deliver to Guest's Room
Return to Room-Service Station
Debit Guest's Account

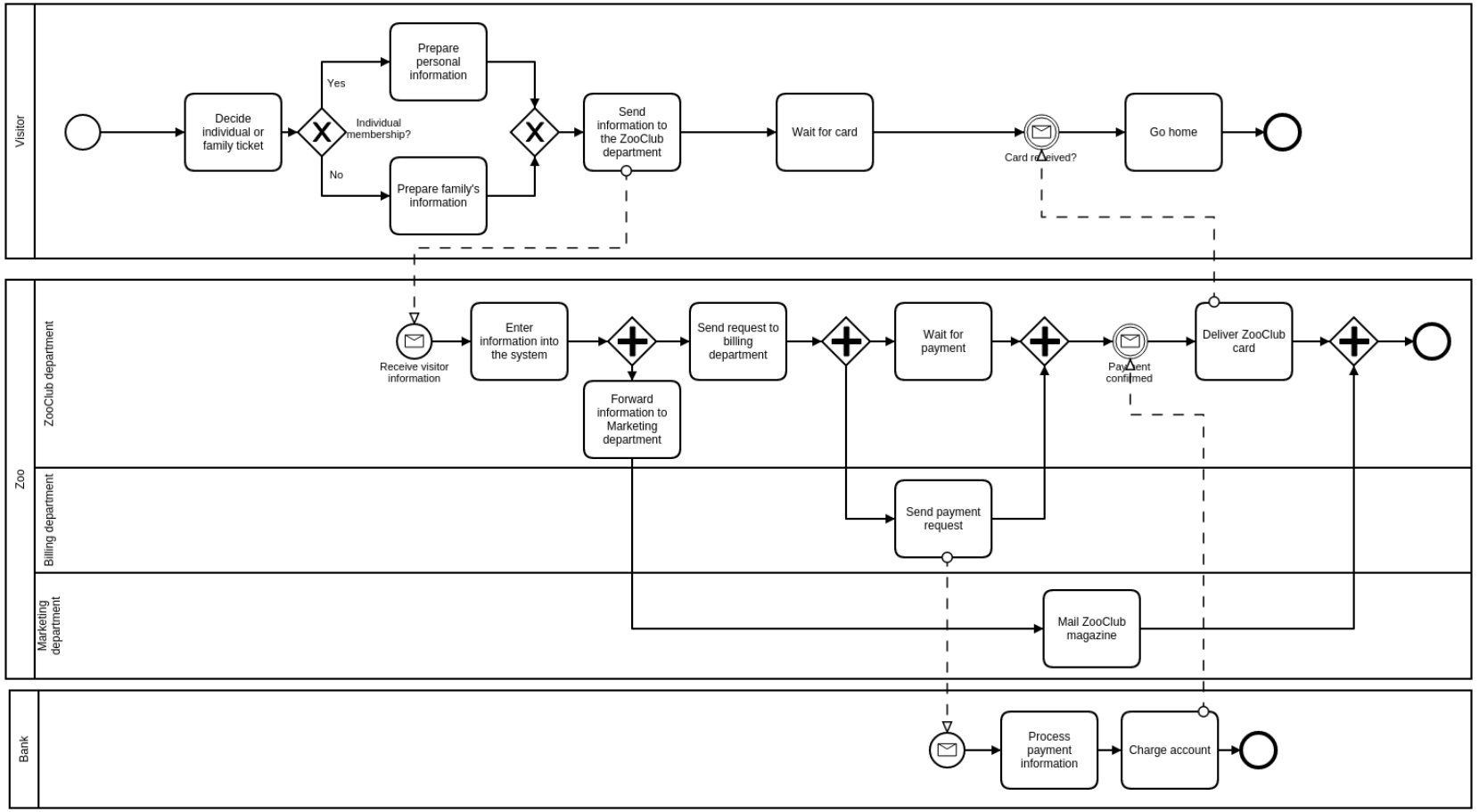
Underwriter [UNWR]



Hospital [HOSP]



Zoo [ZOO]


Visitor
Decide individual or family ticket
Individual membership?
Yes
No
Prepare personal information
Prepare family's information
Send information to the ZooClub department
Wait for card
Card received?
Go home
Zoo
ZooClub department
Receive visitor information
Enter information into the system
Forward information to Marketing department
Send request to billing department
Wait for payment
Payment confirmed
Deliver ZooClub card
Billing department
Send payment request
Marketing department
Mail ZooClub magazine
Bank
Process payment information
Charge account

Apèndix B

Codi font del projecte

El codi font del projecte es pot consultar en aquest repositori de GitHub: `https://github.com/setzer22/bpmn_vs_text`.

Apèndix C

Traces d'execució

En aquest annex es recullen algunes traces de l'execució de l'algorisme. Aquestes traces les genera l'algorisme internament durant l'execució en format de text pla i resulten molt útils per a entendre el perquè de la puntuació d'una instància del problema concret.

El format és, a grans trets, el següent:

- Primer es fa un llistat de la separació de frases i tasques de l'algorisme.
- A continuació es mostra la matriu de similaritat. Les files corresponen a les frases i les columnes a les tasques.
- El següent punt és el threshold que l'algorisme ha calculat automàticament.
- A continuació, es llisten les diferents parelles de frase-tasca que ha trobat l'algorisme en el següent format:
  - Text de la frase
  - Text de la tasca
  - Puntuació de similaritat entre les dues
  - Característiques comunes entre les dues features
- Finalment, s'inclouen aquelles tasques per les quals la puntuació de similaritat era menor al threhsold

El programa inclou l'opció de configuració *verbose* perquè s'inclogui encara més informació a aquesta traça. No obstant, amb aquesta opció, les traces hagués estat massa detallades per incloure-les al document.

C.1 Bicycle Manufacturer (BPMN vs Text)

```
=========
Sentences
=========
Text.0) A small company manufactures customized bicycles .
Text.1) Whenever the sales department receives an order , a new process instance is
    created .
Text.2) A member of the sales department can then reject or accept the order for a
    customized bike .
Text.3) In the former case , the process instance is finished .
Text.4) In the latter case , the storehouse and the engineering department are informed
    .
Text.5) The storehouse immediately processes the part list of the order and checks the
    required quantity of each part .
Text.6) If the part is available in-house , it is reserved .
Text.7) If it is not available , it is back-ordered .
Text.8) This procedure is repeated for each item on the part list .
Text.9) In_the_meantime , the engineering department prepares everything for the
    assembling of the ordered bicycle .
Text.10) If the storehouse has successfully reserved or back-ordered every item of the
    part list and the preparation activity has finished , the engineering department
    assembles the bicycle .
Text.11) Afterwards , the sales department ships the bicycle to the customer and
    finishes the process instance .

=====
Tasks
=====
Task.0) Reserve Part
Task.1) Back-order Part
Task.2) Prepare for Assembling
Task.3) Assemble Bicycle
Task.4) Select Unchecked Part
Task.5) Inform Storehouse and Engineering Department
Task.6) Receive Order
Task.7) Check Part Quantity
Task.8) Ship Bicycle to Customer

================
Similarity Matrix
================
```

0	1	2	3	4	5	6	7	8
0.00	0.00	0.00	0.10	0.00	0.00	0.00	0.00	0.10
0.00	0.00	0.00	0.00	0.00	0.00	0.64	0.00	0.00
0.00	0.00	0.00	0.02	0.00	0.00	0.05	0.00	0.02
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

```
| 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.43 | 0.00 | 0.00 | 0.00 |
| 0.11 | 0.00 | 0.00 | 0.00 | 0.00 | 0.04 | 0.04 | 0.54 | 0.00 |
| 0.11 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 0.11 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 0.00 | 0.00 | 0.37 | 0.22 | 0.00 | 0.00 | 0.00 | 0.00 | 0.04 |
| 0.27 | 0.00 | 0.00 | 0.76 | 0.00 | 0.04 | 0.00 | 0.00 | 0.10 |
| 0.00 | 0.00 | 0.00 | 0.04 | 0.00 | 0.00 | 0.00 | 0.00 | 0.10 |

=========
threshold
=========
0.004621672771494001

=========
Matchings
=========
Sentence 10 :
"If the storehouse has successfully reserved or back-ordered every item of the part
    list and the preparation activity has finished , the engineering department
    assembles the bicycle ."
Task 0 :
"Reserve Part"
Matching score: 0.26833976833976836

    Comon features:

    * Contains the synset "component"
    * Contains an hyponym of "relation"
    * Contains an hyponym of "relation"
    * Contains an hyponym of "abstract_entity"
    * Contains the noun "part"
    * The agent of verb "reserve" contains the noun "storehouse"


Sentence 9 :
"In_the_meantime , the engineering department prepares everything for the assembling of
    the ordered bicycle ."
Task 2 :
"Prepare for Assembling"
Matching score: 0.3695906432748538

    Comon features:

    * Contains the synset "prepare"
    * Contains an hyponym of "learn"
    * Contains the verb "prepare"
    * Contains the action "prepare"
    * The agent of verb "prepare" contains the noun "engineering"
```

```
    * The agent of verb "prepare" contains the noun "department"


Sentence 10 :
"If the storehouse has successfully reserved or back-ordered every item of the part
    list and the preparation activity has finished , the engineering department
    assembles the bicycle ."
Task 3 :
"Assemble Bicycle"
Matching score: 0.7608370702541107

    Comon features:

    * Contains the synset "assemble"
    * Contains the synset "bicycle"
    * Contains an hyponym of "interact"
    * Contains an hyponym of "act"
    * Contains an hyponym of "wheeled_vehicle"
    * Contains an hyponym of "03094503-n:04524313-n"
    * Contains the verb "assemble"
    * Contains the noun "bicycle"
    * Contains the action "assemble"
    * The agent of verb "assemble" contains the noun "engineering"
    * The agent of verb "assemble" contains the noun "department"
    * The agent of verb "bicycle" contains the noun "engineering"
    * The agent of verb "bicycle" contains the noun "department"
    * The direct object of verb "assemble" is "bicycle" (noun)
    * The object of verb "assemble" contains the noun "bicycle"


Sentence 4 :
"In the latter case , the storehouse and the engineering department are informed ."
Task 5 :
"Inform Storehouse and Engineering Department"
Matching score: 0.42874692874692877

    Comon features:

    * Contains the synset "depot"
    * Contains the synset "inform"
    * Contains an hyponym of "deposit"
    * Contains an hyponym of "facility"
    * Contains an hyponym of "communicate"
    * Contains an hyponym of "interact"
    * Contains the noun "storehouse"
    * Contains the verb "inform"
    * Contains the action "inform"
    * The direct object of verb "inform" is "storehouse" (noun)
    * The object of verb "inform" contains the noun "storehouse"
```

```
Sentence 1 :
"Whenever the sales department receives an order , a new process instance is created ."
Task 6 :
"Receive Order"
Matching score: 0.6404494382022472

    Comon features:

    * Contains the synset "have"
    * Contains the synset "order"
    * Contains an hyponym of "acquire"
    * Contains an hyponym of "organisation"
    * Contains an hyponym of "activity"
    * Contains an hyponym of "activity"
    * Contains the verb "receive"
    * Contains the noun "order"
    * Contains the action "receive"
    * The agent of verb "receive" contains the noun "sale"
    * The agent of verb "receive" contains the noun "department"
    * The direct object of verb "receive" is "order" (noun)
    * The object of verb "receive" contains the noun "order"


Sentence 5 :
"The storehouse immediately processes the part list of the order and checks the
   required quantity of each part ."
Task 7 :
"Check Part Quantity"
Matching score: 0.5412844036697247

    Comon features:

    * Contains the synset "bank_check"
    * Contains an hyponym of "bill_of_exchange"
    * Contains an hyponym of "negotiable_instrument"
    * Contains the noun "check"


Sentence 11 :
"Afterwards , the sales department ships the bicycle to the customer and finishes the
   process instance ."
Task 8 :
"Ship Bicycle to Customer"
Matching score: 0.10249307479224377

    Comon features:
```

```
    * Contains the synset "bicycle"
    * Contains the synset "client"
    * Contains an hyponym of "wheeled_vehicle"
    * Contains an hyponym of "03094503-n:04524313-n"
    * Contains an hyponym of "consumer"
    * Contains an hyponym of "user"
    * Contains the noun "bicycle"
    * Contains the particle "to"
    * Contains the noun "customer"
```

```
=========================
Tasks with no good match
=========================
Task 1 :
"Back-order Part"
Task 4 :
"Select Unchecked Part"
```

C.2 Dispatch of Goods (BPMN vs Text)

```
=========
Sentences
=========
Text.0) If goods shall be shipped , the secretary clarifies who will do the shipping .
Text.1) If you have large amounts , special shipping will be necessary .
Text.2) In these cases the secretary invites three logistic companies to make offers
   and she selects one of them .
Text.3) In_case of small amounts , normal post shipment is used .
Text.4) Therefore a package label is written by the secretary and a parcel insurance
   taken by the logistics department head if necessary .
Text.5) In_the_meantime the goods can be already packaged by the warehousemen .
Text.6) If everything is ready , the packaged goods are prepared for being picked up by
    the logistic company .

=====
Tasks
=====
Task.0) Package goods
Task.1) Get 3 offers
from logistic
companies
Task.2) Write package
label
Task.3) Select logistic
company and
place order
```

```
Task.4) Prepare for
picking up
goods
Task.5) Clarify shipment method
Task.6) Insure parcel

==================
Similarity Matrix
==================

|    0 |    1 |    2 |    3 |    4 |    5 |    6 |
|------+------+------+------+------+------+------|
| 0.50 | 0.00 | 0.00 | 0.00 | 0.03 | 0.16 | 0.00 |
| 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 0.00 | 0.14 | 0.00 | 0.10 | 0.00 | 0.00 | 0.00 |
| 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.02 | 0.00 |
| 0.50 | 0.00 | 0.79 | 0.00 | 0.00 | 0.00 | 0.06 |
| 0.50 | 0.00 | 0.00 | 0.00 | 0.04 | 0.00 | 0.00 |
| 0.50 | 0.07 | 0.00 | 0.10 | 0.45 | 0.00 | 0.00 |

=========
threshold
=========
0.043222326538446015

=========
Matchings
=========
Sentence 0 :
"If goods shall be shipped , the secretary clarifies who will do the shipping ."
Task 0 :
"Package goods"
Matching score: 0.5

    Comon features:

    * Contains the synset "commodity"
    * Contains an hyponym of "artefact"
    * Contains an hyponym of "unit"
    * Contains the noun "good"


Sentence 2 :
"In these cases the secretary invites three logistic companies to make offers and she
    selects one of them ."
Task 1 :
"Get 3 offers\nfrom logistic\ncompanies"
Matching score: 0.1423125794155019
```

```
    Comon features:

    * Contains the synset "logistic"
    * Contains the synset "company"
    * Contains the synset "get"
    * Contains the synset "offer"
    * Contains an hyponym of "establishment"
    * Contains an hyponym of "organisation"
    * Contains an hyponym of "alter"
    * Contains an hyponym of "content"
    * Contains an hyponym of "communication"
    * Contains the number "3"
    * Contains the adjective "logistic"
    * Contains the noun "company"
    * Contains the noun "offer"


Sentence 4 :
"Therefore a package label is written by the secretary and a parcel insurance taken by
   the logistics department head if necessary ."
Task 2 :
"Write package\nlabel"
Matching score: 0.7925669835782195

    Comon features:

    * Contains the synset "package"
    * Contains the synset "label"
    * Contains the synset "write"
    * Contains the synset "package"
    * Contains an hyponym of "container"
    * Contains an hyponym of "instrumentality"
    * Contains an hyponym of "mark"
    * Contains an hyponym of "symbol"
    * Contains an hyponym of "communicate"
    * Contains an hyponym of "interact"
    * Contains an hyponym of "container"
    * Contains an hyponym of "instrumentality"
    * Contains the noun "package"
    * Contains the noun "label"
    * Contains the verb "write"
    * Contains the action "write"
    * The agent of verb "write" contains the noun "secretary"
    * The direct object of verb "write" is "label" (noun)
    * The object of verb "write" contains the noun "package"
    * The object of verb "write" contains the noun "label"


Sentence 6 :
```

```
"If everything is ready , the packaged goods are prepared for being picked up by the
    logistic company ."
Task 3 :
"Select logistic\ncompany and\nplace order"
Matching score: 0.10148975791433891

    Comon features:

    * Contains the synset "logistic"
    * Contains the synset "company"
    * Contains an hyponym of "establishment"
    * Contains an hyponym of "organisation"
    * Contains an hyponym of "organisation"
    * Contains the adjective "logistic"
    * Contains the noun "company"


Sentence 6 :
"If everything is ready , the packaged goods are prepared for being picked up by the
    logistic company ."
Task 4 :
"Prepare for\npicking up\ngoods"
Matching score: 0.4505494505494506

    Comon features:

    * Contains the synset "commodity"
    * Contains the synset "prepare"
    * Contains the synset "pick"
    * Contains the synset "up"
    * Contains an hyponym of "artefact"
    * Contains an hyponym of "unit"
    * Contains an hyponym of "learn"
    * Contains an hyponym of "choose"
    * Contains an hyponym of "decide"
    * Contains the noun "good"
    * Contains the verb "prepare"
    * Contains the verb "pick"
    * Contains the particle "up"
    * Contains the action "prepare"
    * Contains the action "pick"
    * The direct object of verb "pick" is "good" (noun)
    * The object of verb "prepare" contains the noun "good"
    * The object of verb "pick" contains the noun "good"


Sentence 0 :
"If goods shall be shipped , the secretary clarifies who will do the shipping ."
Task 5 :
```

```
"Clarify shipment method"
Matching score: 0.16181448882870683

    Comon features:

    * Contains the synset "clarify"
    * Contains an hyponym of "explain"
    * Contains an hyponym of "inform"
    * Contains the verb "clarify"
    * Contains the action "clarify"
    * The agent of verb "clarify" contains the noun "secretary"


Sentence 4 :
"Therefore a package label is written by the secretary and a parcel insurance taken by
    the logistics department head if necessary ."
Task 6 :
"Insure parcel"
Matching score: 0.05795677799607073

    Comon features:

    * Contains the synset "package"
    * Contains the synset "package"
    * Contains an hyponym of "container"
    * Contains an hyponym of "instrumentality"
    * Contains an hyponym of "container"
    * Contains an hyponym of "instrumentality"
    * Contains the noun "parcel"



==========
Tasks with no good match
==========
```

C.3 Hotel (BPMN vs Text)

```
========
Sentences
========
Text.0) The Evanstonian is an upscale independent hotel .
Text.1) When a guest calls room service at The_Evanstonian , the room-service manager
    takes down the order .
Text.2) She then submits an order ticket to the kitchen to begin preparing the food .
Text.3) She also gives an order to the sommelier ( i.e. , the wine waiter ) to fetch
    wine from the cellar and to prepare any other alcoholic beverages .
Text.4) Eighty percent of room-service orders include wine or some other alcoholic
```

```
    beverage .
Text.5) Finally , she assigns the order to the waiter .
Text.6) While the kitchen and the sommelier are doing their tasks , the waiter readies
    a cart ( i.e. , puts a tablecloth on the cart and gathers silverware ) .
Text.7) The waiter is also responsible for nonalcoholic drinks .
Text.8) Once the food , wine , and cart are ready , the waiter delivers it to the guest
    ? s room .
Text.9) After returning to the room-service station , the waiter debits the guest ? s
    account .
Text.10) The waiter may wait to do the billing if he has another order to prepare or
    deliver .

=====
Tasks
=====
Task.0) Prepare Non-Alcoholic Drinks
Task.1) Fetch Wine from Cellar
Task.2) Return to Room-Service Station
Task.3) Prepare Meal
Task.4) Debit Guest's Account
Task.5) Ready Cart
Task.6) Deliver to Guest's Room
Task.7) Submit Order Ticket to Kitchen
Task.8) Take down Order
Task.9) Assign Order to Waiter
Task.10) Prepare Alcoholic Beverages
Task.11) Give Order to Sommelier

================
Similarity Matrix
================
```

0	1	2	3	4	5	6	7	8	9	10	11
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.35	0.05	0.00	0.05
0.16	0.00	0.22	0.28	0.00	0.00	0.08	0.26	0.16	0.07	0.19	0.07
0.24	0.83	0.22	0.19	0.00	0.00	0.08	0.03	0.11	0.12	0.19	0.61
0.00	0.06	0.00	0.00	0.00	0.00	0.00	0.00	0.11	0.05	0.00	0.05
0.00	0.00	0.22	0.00	0.00	0.00	0.08	0.03	0.11	0.61	0.00	0.07
0.03	0.00	0.00	0.03	0.00	0.08	0.01	0.06	0.00	0.05	0.03	0.05
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.05	0.00	0.00
0.00	0.07	0.22	0.00	0.00	0.15	0.67	0.03	0.00	0.07	0.00	0.02
0.00	0.00	0.22	0.00	0.00	0.00	0.08	0.03	0.00	0.07	0.00	0.02
0.16	0.00	0.24	0.19	0.00	0.00	0.47	0.03	0.11	0.12	0.19	0.07

```
=========
threshold
=========
```

0.02935206128863406

==========
Matchings
==========
Sentence 3 :
"She also gives an order to the sommelier (i.e. , the wine waiter) to fetch wine from the cellar and to prepare any other alcoholic beverages ."
Task 1 :
"Fetch Wine from Cellar"
Matching score: 0.8255179934569248

Comon features:

* Contains the synset "vino"
* Contains the synset "bring"
* Contains the synset "vino"
* Contains the synset "basement"
* Contains an hyponym of "alcohol"
* Contains an hyponym of "03248958-n:07881800-n"
* Contains an hyponym of "channel"
* Contains an hyponym of "displace"
* Contains an hyponym of "alcohol"
* Contains an hyponym of "03248958-n:07881800-n"
* Contains an hyponym of "floor"
* Contains an hyponym of "construction"
* Contains the noun "wine"
* Contains the verb "fetch"
* Contains the noun "wine"
* Contains the noun "cellar"
* Contains the action "fetch"
* The direct object of verb "fetch" is "wine" (noun)
* The object of verb "fetch" contains the noun "wine"

Sentence 10 :
"The waiter may wait to do the billing if he has another order to prepare or deliver ."
Task 2 :
"Return to Room-Service Station"
Matching score: 0.23741007194244604

Comon features:

* Contains an hyponym of "speech_act"
* Contains the particle "to"
* Contains the particle "to"

Sentence 2 :

```
"She then submits an order ticket to the kitchen to begin preparing the food ."
Task 3 :
"Prepare Meal"
Matching score: 0.2785547785547786

    Comon features:

    * Contains the synset "cook"
    * Contains an hyponym of "create_from_raw_material"
    * Contains an hyponym of "create"
    * Contains the verb "prepare"
    * Contains the action "prepare"
    * The agent of verb "prepare" contains the noun "kitchen"


Sentence 8 :
"Once the food , wine , and cart are ready , the waiter delivers it to the guest ? s
   room ."
Task 5 :
"Ready Cart"
Matching score: 0.14552736982643524

    Comon features:

    * Contains the synset "cart"
    * Contains the synset "ready"
    * Contains an hyponym of "waggon"
    * Contains an hyponym of "wheeled_vehicle"
    * Contains the noun "cart"
    * Contains the adjective "ready"


Sentence 8 :
"Once the food , wine , and cart are ready , the waiter delivers it to the guest ? s
   room ."
Task 6 :
"Deliver to Guest's Room"
Matching score: 0.674185463659148

    Comon features:

    * Contains the synset "bear"
    * Contains an hyponym of "bring_forth"
    * Contains an hyponym of "create"
    * Contains the verb "deliver"
    * Contains the particle "to"
    * Contains the action "deliver"
    * The agent of verb "deliver" contains the noun "waiter"
```

```
Sentence 2 :
"She then submits an order ticket to the kitchen to begin preparing the food ."
Task 7 :
"Submit Order Ticket to Kitchen"
Matching score: 0.26439232409381663

    Comon features:

    * Contains the synset "submit"
    * Contains the synset "kitchen"
    * Contains an hyponym of "undergo"
    * Contains an hyponym of "change"
    * Contains an hyponym of "room"
    * Contains an hyponym of "area"
    * Contains the verb "submit"
    * Contains the particle "to"
    * Contains the noun "kitchen"
    * Contains the particle "to"
    * Contains the action "submit"


Sentence 1 :
"When a guest calls room service at The_Evanstonian , the room-service manager takes
    down the order ."
Task 8 :
"Take down Order"
Matching score: 0.35130111524163565

    Comon features:

    * Contains the synset "order"
    * Contains an hyponym of "commercial_document"
    * Contains an hyponym of "document"
    * Contains the verb "take"
    * Contains the noun "order"
    * Contains the action "take"


Sentence 5 :
"Finally , she assigns the order to the waiter ."
Task 9 :
"Assign Order to Waiter"
Matching score: 0.6105919003115264

    Comon features:

    * Contains the synset "assign"
    * Contains the synset "order"
```

```
    * Contains the synset "server"
    * Contains an hyponym of "apply"
    * Contains an hyponym of "commercial_document"
    * Contains an hyponym of "document"
    * Contains an hyponym of "dining-room_attendant"
    * Contains an hyponym of "employee"
    * Contains the verb "assign"
    * Contains the noun "order"
    * Contains the particle "to"
    * Contains the noun "waiter"
    * Contains the action "assign"
    * The direct object of verb "assign" is "order" (noun)
    * The object of verb "assign" contains the noun "order"


Sentence 3 :
"She also gives an order to the sommelier ( i.e. , the wine waiter ) to fetch wine from
    the cellar and to prepare any other alcoholic beverages ."
Task 10 :
"Prepare Alcoholic Beverages"
Matching score: 0.18727915194346292

    Comon features:

    * Contains the synset "cook"
    * Contains an hyponym of "create_from_raw_material"
    * Contains an hyponym of "create"
    * Contains the verb "prepare"
    * Contains the action "prepare"


Sentence 3 :
"She also gives an order to the sommelier ( i.e. , the wine waiter ) to fetch wine from
    the cellar and to prepare any other alcoholic beverages ."
Task 11 :
"Give Order to Sommelier"
Matching score: 0.6105919003115265

    Comon features:

    * Contains the synset "generate"
    * Contains the synset "order"
    * Contains the synset "sommelier"
    * Contains an hyponym of "create"
    * Contains an hyponym of "commercial_document"
    * Contains an hyponym of "document"
    * Contains an hyponym of "server"
    * Contains an hyponym of "dining-room_attendant"
    * Contains an hyponym of "dining-room_attendant"
```

```
    * Contains the verb "give"
    * Contains the noun "order"
    * Contains the particle "to"
    * Contains the noun "sommelier"
    * Contains the particle "to"
    * Contains the particle "to"
    * Contains the action "give"
    * The direct object of verb "give" is "order" (noun)
    * The object of verb "give" contains the noun "order"
```

```
==========================
Tasks with no good match
==========================
Task 0 :
"Prepare Non-Alcoholic Drinks"
Task 4 :
"Debit Guest's Account"
```

C.4 Zoo (BPMN vs Text)

```
=========
Sentences
=========
Text.0) When a visitor wants to become a member of Barcelona_\s_ZooClub , the following
    steps must be taken .
Text.1) First of all , the customer must decide whether he wants an individual or
   family membership .
Text.2) If he wants an individual membership , he must prepare his personal information
    .
Text.3) If he wants a family membership instead , he should prepare the information for
    its spouse and spawn as_well .
Text.4) The customer must then give this information to the ZooClub department .
Text.5) The ZooClub department introduces the visitor \s personal data into the system
   and takes the payment request to the Billing department .
Text.6) The ZooClub department also forwards the visitor \s information to the
   marketing department .
Text.7) The billing department sends the payment request to the bank .
Text.8) The bank processes the payment information and , if everything is correct ,
   charges the payment into user \s account .
Text.9) Once the payment is confirmed , the ZooClub department can print the card and
   deliver it to the visitor .
Text.10) In_the_meantime , the Marketing department makes a request to mail the
   Zoo_Club \s magazine to the visitor \s home .
Text.11) Once the visitor receives the card , he can go home .

=====
```

Tasks

Task.0) Deliver ZooClub card
Task.1) Process payment information
Task.2) Prepare personal information
Task.3) Mail ZooClub magazine
Task.4) Send payment request
Task.5) Charge account
Task.6) Enter information into the system
Task.7) Go home
Task.8) Wait for card
Task.9) Send information to the ZooClub department
Task.10) Send request to billing department
Task.11) Prepare family's information
Task.12) Decide individual or family ticket
Task.13) Forward information to Marketing department
Task.14) Wait for payment

Similarity Matrix

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.01	0.00	0.00	0.06	0.00
0.00	0.00	0.00	0.00	0.00	0.00	0.10	0.00	0.00	0.00	0.05	0.40	0.00	0.00	
0.00	0.23	0.75	0.00	0.00	0.00	0.06	0.00	0.00	0.06	0.00	0.56	0.05	0.11	0.00
0.00	0.23	0.63	0.00	0.00	0.00	0.06	0.10	0.00	0.06	0.00	0.60	0.06	0.11	0.00
0.03	0.23	0.06	0.07	0.00	0.00	0.06	0.00	0.00	0.19	0.06	0.05	0.00	0.29	0.00
0.03	0.35	0.07	0.07	0.24	0.04	0.09	0.00	0.00	0.16	0.08	0.05	0.00	0.23	0.05
0.03	0.23	0.06	0.07	0.00	0.00	0.06	0.00	0.00	0.20	0.04	0.07	0.00	0.40	0.00
0.00	0.24	0.00	0.00	0.59	0.04	0.00	0.00	0.00	0.27	0.44	0.00	0.00	0.17	0.05
0.00	0.46	0.06	0.00	0.21	0.18	0.06	0.00	0.00	0.06	0.00	0.07	0.00	0.11	0.05
0.38	0.23	0.00	0.07	0.02	0.00	0.00	0.00	0.08	0.13	0.04	0.00	0.00	0.17	0.05
0.00	0.01	0.00	0.14	0.07	0.00	0.00	0.20	0.00	0.13	0.30	0.05	0.03	0.29	0.00
0.06	0.00	0.00	0.00	0.00	0.00	0.00	0.65	0.08	0.00	0.00	0.02			

```
    0.03 | 0.00 | 0.00 |

=========
threshold
=========
0.032056243453235655

=========
Matchings
=========
Sentence 9 :
"Once the payment is confirmed , the ZooClub department can print the card and deliver
    it to the visitor ."
Task 0 :
"Deliver ZooClub card"
Matching score: 0.3820224719101124

    Comon features:

    * Contains the synset "card"
    * Contains the synset "deliver"
    * Contains an hyponym of "positive_identification"
    * Contains an hyponym of "identification"
    * Contains an hyponym of "bring"
    * Contains an hyponym of "channel"
    * Contains the noun "zooclub"
    * Contains the noun "card"
    * Contains the verb "deliver"
    * Contains the action "deliver"
    * The agent of verb "deliver" contains the noun "zooclub"
    * The agent of verb "deliver" contains the noun "department"


Sentence 8 :
"The bank processes the payment information and , if everything is correct , charges
    the payment into user \\s account ."
Task 1 :
"Process payment information"
Matching score: 0.4591439688715953

    Comon features:

    * Contains the synset "payment"
    * Contains the synset "data"
    * Contains the synset "payment"
    * Contains an hyponym of "cost"
    * Contains an hyponym of "expenditure"
    * Contains an hyponym of "accumulation"
    * Contains an hyponym of "group"
```

```
    * Contains an hyponym of "cost"
    * Contains an hyponym of "expenditure"
    * Contains the noun "payment"
    * Contains the noun "information"
    * Contains the noun "payment"


Sentence 2 :
"If he wants an individual membership , he must prepare his personal information ."
Task 2 :
"Prepare personal information"
Matching score: 0.7502347417840375

    Comon features:

    * Contains the synset "personal"
    * Contains the synset "data"
    * Contains an hyponym of "accumulation"
    * Contains an hyponym of "group"
    * Contains the verb "prepare"
    * Contains the adjective "personal"
    * Contains the noun "information"
    * Contains the action "prepare"
    * The direct object of verb "prepare" is "information" (noun)
    * The object of verb "prepare" contains the adjective "personal"
    * The object of verb "prepare" contains the noun "information"


Sentence 10 :
"In_the_meantime , the Marketing department makes a request to mail the Zoo_Club \\s
    magazine to the visitor \\s home ."
Task 3 :
"Mail ZooClub magazine"
Matching score: 0.1382716049382716

    Comon features:

    * Contains the synset "mag"
    * Contains an hyponym of "06263369-n:06589574-n"
    * Contains the noun "magazine"


Sentence 7 :
"The billing department sends the payment request to the bank ."
Task 4 :
"Send payment request"
Matching score: 0.5868096940802543

    Comon features:
```

```
    * Contains the synset "mail"
    * Contains the synset "payment"
    * Contains the synset "asking"
    * Contains an hyponym of "speech_act"
    * Contains an hyponym of "transfer"
    * Contains an hyponym of "displace"
    * Contains an hyponym of "cost"
    * Contains an hyponym of "expenditure"
    * Contains an hyponym of "speech_act"
    * Contains an hyponym of "act"
    * Contains the verb "send"
    * Contains the noun "payment"
    * Contains the noun "request"
    * Contains the action "send"
    * The agent of verb "send" contains the noun "billing"
    * The agent of verb "send" contains the noun "department"
    * The direct object of verb "send" is "request" (noun)
    * The direct object of verb "request" is "payment" (noun)
    * The object of verb "send" contains the noun "payment"
    * The object of verb "send" contains the noun "request"
    * The object of verb "request" contains the noun "payment"


Sentence 8 :
"The bank processes the payment information and , if everything is correct , charges
   the payment into user \\s account ."
Task 5 :
"Charge account"
Matching score: 0.18337730870712401

    Comon features:

    * Contains the synset "account"
    * Contains an hyponym of "financial_statement"
    * Contains an hyponym of "commercial_document"
    * Contains the noun "account"
    * The agent of verb "charge" contains the noun "bank"


Sentence 5 :
"The ZooClub department introduces the visitor \\s personal data into the system and
   takes the payment request to the Billing department ."
Task 6 :
"Enter information into the system"
Matching score: 0.08826479438314946

    Comon features:
```

```
    * Contains the synset "data"
    * Contains the synset "system"
    * Contains an hyponym of "accumulation"
    * Contains an hyponym of "group"
    * Contains an hyponym of "matter"
    * Contains an hyponym of "physical_entity"
    * Contains the noun "system"


Sentence 11 :
"Once the visitor receives the card , he can go home ."
Task 7 :
"Go home"
Matching score: 0.6539792387543253

    Comon features:

    * Contains the synset "family"
    * Contains an hyponym of "social_unit"
    * Contains an hyponym of "organisation"
    * Contains the verb "go"
    * Contains the noun "home"
    * Contains the action "go"


Sentence 11 :
"Once the visitor receives the card , he can go home ."
Task 8 :
"Wait for card"
Matching score: 0.07672301690507152

    Comon features:

    * Contains the synset "card"
    * Contains an hyponym of "positive_identification"
    * Contains an hyponym of "identification"
    * Contains the noun "card"


Sentence 4 :
"The customer must then give this information to the ZooClub department ."
Task 9 :
"Send information to the ZooClub department"
Matching score: 0.19411123227917124

    Comon features:

    * Contains the synset "data"
    * Contains the synset "department"
```

```
    * Contains an hyponym of "accumulation"
    * Contains an hyponym of "group"
    * Contains an hyponym of "division"
    * Contains an hyponym of "administrative_body"
    * Contains the noun "information"
    * Contains the particle "to"
    * Contains the noun "zooclub"
    * Contains the noun "department"


Sentence 7 :
"The billing department sends the payment request to the bank ."
Task 10 :
"Send request to billing department"
Matching score: 0.4403892944038929

    Comon features:

    * Contains the synset "billing"
    * Contains the synset "department"
    * Contains the synset "mail"
    * Contains the synset "asking"
    * Contains an hyponym of "asking"
    * Contains an hyponym of "speech_act"
    * Contains an hyponym of "speech_act"
    * Contains an hyponym of "division"
    * Contains an hyponym of "administrative_body"
    * Contains an hyponym of "transfer"
    * Contains an hyponym of "displace"
    * Contains an hyponym of "speech_act"
    * Contains an hyponym of "speech_act"
    * Contains an hyponym of "act"
    * Contains the noun "billing"
    * Contains the noun "department"
    * Contains the verb "send"
    * Contains the noun "request"
    * Contains the particle "to"
    * Contains the action "send"
    * The agent of verb "billing" contains the noun "department"
    * The agent of verb "send" contains the noun "department"
    * The direct object of verb "send" is "request" (noun)
    * The object of verb "send" contains the noun "request"
    * The object of verb "send" contains the particle "to"


Sentence 3 :
"If he wants a family membership instead , he should prepare the information for its
   spouse and spawn as_well ."
Task 11 :
```

```
"Prepare family's information"
Matching score: 0.6046511627906976

    Comon features:

    * Contains the synset "family"
    * Contains the synset "data"
    * Contains an hyponym of "social_unit"
    * Contains an hyponym of "organisation"
    * Contains an hyponym of "accumulation"
    * Contains an hyponym of "group"
    * Contains the noun "family"
    * Contains the verb "prepare"
    * Contains the noun "information"
    * Contains the action "prepare"
    * The direct object of verb "prepare" is "information" (noun)
    * The object of verb "prepare" contains the noun "information"


Sentence 1 :
"First of all , the customer must decide whether he wants an individual or family
    membership ."
Task 12 :
"Decide individual or family ticket"
Matching score: 0.4036608863198459

    Comon features:

    * Contains the synset "decide"
    * Contains the synset "individual"
    * Contains the synset "family"
    * Contains an hyponym of "social_unit"
    * Contains an hyponym of "organisation"
    * Contains the verb "decide"
    * Contains the adjective "individual"
    * Contains the noun "family"
    * Contains the action "decide"
    * The object of verb "decide" contains the adjective "individual"
    * The object of verb "decide" contains the noun "family"


Sentence 6 :
"The ZooClub department also forwards the visitor \\s information to the marketing
    department ."
Task 13 :
"Forward information to Marketing department"
Matching score: 0.40038684719535783

    Comon features:
```

```
    * Contains the synset "department"
    * Contains the synset "data"
    * Contains the synset "marketing"
    * Contains the synset "department"
    * Contains an hyponym of "division"
    * Contains an hyponym of "administrative_body"
    * Contains an hyponym of "accumulation"
    * Contains an hyponym of "group"
    * Contains an hyponym of "commerce"
    * Contains an hyponym of "dealing"
    * Contains an hyponym of "division"
    * Contains an hyponym of "administrative_body"
    * Contains the noun "department"
    * Contains the noun "information"
    * Contains the particle "to"
    * Contains the noun "marketing"
    * Contains the noun "department"


Sentence 9 :
"Once the payment is confirmed , the ZooClub department can print the card and deliver
    it to the visitor ."
Task 14 :
"Wait for payment"
Matching score: 0.0504704875962361

    Comon features:

    * Contains the synset "payment"
    * Contains an hyponym of "cost"
    * Contains an hyponym of "expenditure"
    * Contains the noun "payment"



==========
Tasks with no good match
==========
```