Extraction et catégorisation de l'information temporelle de textes scientifiques

Salah Yahiaoui

THÈSE DE DOCTORAT DE L'ÉTABLISSEMENT UNIVERSITÉ BOURGOGNE FRANCHE-COMTÉ

PRÉPARE À UNIVERSITÉ DE FRANCHE-COMTÉ

Ecole doctorale n° 592

ÉCOLE DOCTORALE LECLA

Doctorat en Sciences du Langage, mention Traitement Automatique des Langues

Par

Salah YAHIAOUI

Sous la direction de Iana ATANASSOVA

Extraction et catégorisation de l'information temporelle de textes scientifiques

Thèse présentée et soutenue à Besançon, le 08 décembre 2023

Composition du Jury :

Professeur Cyril LABBÉ	LIG, Université Grenoble Alpes	Rapporteur
Professeur Guillaume CABANAC	IRIT, Université Toulouse ; IUF	Rapporteur
Professeure émérite Sylviane CARDEY	CRIT, Université de Franche-Comté ; IUF	Examinatrice
Professeur émérite Mohamed HASSOUN	ENSSIB	Président
Dr. Marc BERTIN	ELICO, Université Lyon 1	Examinateur
Dr. Iana ATANASSOVA	CRIT, Université de Franche-Comté ; IUF	Directrice de thèse

LECLA
ÉCOLE DOCTORALE

Titre : Extraction et catégorisation d'informations temporelles de textes scientifiques

Mots clés : TimeInfo, extraction d'information, information temporelle, catégorisation sémantique, articles scientifiques, TAL

Résumé : Cette thèse aborde la problématique du traitement de corpus scientifiques, d'un point de vue linguistique, afin d'en extraire, catégoriser et agréger les informations spatio-temporelles pour produire de nouvelles représentations de l'information textuelle. Dans un premier temps, nous proposons le schéma d'annotation TimeInfo, qui permet de rendre compte de la sémantique des différentes expressions temporelles dans les textes scientifiques. Nous montrons l'apport de TimeInfo par rapport aux schémas d'annotation existants, notamment TimeML. Dans un deuxième temps, nous construisons des ensembles de règles linguistiques pour l'annotation automatique des corpus scientifiques avec TimeInfo. Nous traitons le corpus CORD-19 et produisons un nouveau corpus annoté, TimeTank. Enfin, nous proposons des applications autour de TimeInfo et abordons la problématique des informations spatiales, par une expérimentation sur leur annotation et cartographie.

Title: Extraction and categorization of temporal information from scientific texts

Keywords : TimeInfo, Information Extraction, temporal information, semantic classification, scientific papers, NLP

Abstract: This thesis addresses the problem of processing scientific corpora from a linguistic point of view in order to extract, categorise and aggregate spatio-temporal information in order to produce new representations of textual information. First, we propose the TimeInfo annotation scheme, which allows us to take into account the thematic nature of different temporal expressions in scientific texts. We show the contribution of TimeInfo compared to existing annotation schemes, in particular TimeML. Secondly, we construct sets of linguistic rules for the automatic annotation of scientific corpora with TimeInfo. We process the CORD-19 corpus and produce a new annotated corpus TimeTank. Finally, we propose applications based on TimeInfo and address the problem of spatial information by experimenting with its annotation and mapping.

UBFC
UNIVERSITÉ BOURGOGNE FRANCHE-COMTE

Université Bourgogne Franche-Comté
32, avenue de l'Observatoire
25000 Besancon

“Be the artist who creates with fervor, even when no eyes bear witness.”

— Salah YAHIAOUI

À ma mère, Sadia

et

À mon père, Arezki

Remerciement

Une thèse est bien plus qu'un simple travail académique ; c'est une aventure. Et comme toute aventure, elle n'a pas seulement ses moments de gloire. Dans les périodes difficiles, j'ai toujours pu compter sur ma directrice. Son soutien, sa sagesse, son expertise, sa rigueur exemplaire et ses encouragements m'ont donné la force et l'envie de mener à bien ma thèse de doctorat. Du plus profond de mon être, merci Dr. Iana Atanassova.

Je tiens à exprimer ma profonde gratitude à mon comité de suivi de thèse et aux honorables membres du jury. Mes remerciements vont en particulier au Prof. Sylviane Cardey, au Prof. Cyril Labbé, au Prof. Guillaume Cabanac, au Prof. Mohammed Hassoun, au Dr. Marc Bertin et au Dr. Marion Bendinelli.

Je tiens à adresser une mention toute particulière au Dr. Izabella Thomas, au Dr. Margaret Gillespie, au Dr. Philippe Laplace. Votre humanité et votre dévouement sont une véritable source d'inspiration. Grâce à vous, j'ai trouvé le courage de persévérer et de toujours avancer. Du fond du cœur, merci.

À mes chers parents, à mon cher frère Zahir, à mes chères sœurs Malika, Lynda et Saliha. Les mots peinent à capturer l'immensité de ma gratitude. Votre présence constante, votre soutien, vos encouragements sincères et votre aide précieuse m'ont accompagné à chaque étape de ma vie. Je suis profondément honoré et fier d'être votre fils et votre frère.

À Lamine, mon ami de toujours, les années d'amitié que nous avons partagées sont inestimables. Ta présence constante et rassurante m'ont été d'une aide inestimable, merci pour tout. Je souhaite aussi exprimer ma profonde gratitude à l'ensemble de mes amis qui ont été des piliers de soutien et d'encouragement. Mes remerciements spécifiques vont à Marina, Amel, Massinissa, Mohamed Amine, Samy, Samir et Rafik.

Enfin, il serait impensable de conclure ces remerciements sans évoquer mes collègues qui ont enrichi mon parcours. Une pensée chaleureuse pour François, Youcef, Yagmur, Nicolas, Aurélie et Ningrum.

Table des matières

Table des figures

Liste des tableaux

Introduction

Cette thèse aborde la problématique du traitement de corpus scientifiques, d'un point de vue linguistique, afin d'en extraire, catégoriser et agréger les informations spatio-temporelles pour produire de nouvelles représentations de l'information textuelle. Nous nous focalisons sur les informations temporelles, leur analyse, extraction et catégorisation. Si des standards d'annotation existent dans ce domaine, notamment TIMEX3 de TimeML, notre travail permet d'aller plus loin dans les distinctions entre les types d'intervalles temporels et de prendre en compte des expressions temporelles complexes qui ne sont pas représentées et traitées par les systèmes actuels.

Nos objectifs s'articulent autour de plusieurs axes. Dans un premier temps, nous proposons le schéma d'annotation TimeInfo, qui permet de rendre compte de la sémantique des différentes expressions temporelles dans les textes scientifiques. Nous montrons l'apport de TimeInfo par rapport aux schémas d'annotation existants. Dans un deuxième temps, nous proposons des ensembles de règles linguistiques pour l'annotation automatique des corpus et construisons un corpus annoté TimeTank [Yahiaoui and Atanassova, 2023], disponible en accès libre[1]. Nous proposons des applications autour de TimeInfo. Enfin, nous abordons la problématique des informations spatiales, par une expérimentation sur leur annotation et cartographie.

Les analyses sémantiques de textes scientifiques permettent de produire de nouvelles métadonnées liées aux textes, par l'extraction et la catégorisation d'informations spatiales

[1] https://zenodo.org/record/8364409

ou temporelles dans un processus de gestion de données. Ces métadonnées pourront servir comme support pour la production de nouvelles représentations sous forme graphique ou sous forme de synthèse textuelles, avec des applications en cartographie, en analyse chronologique et visualisations. Les informations spatiales et temporelles contenues dans les corpus scientifiques sont une source d'information qui dépasse le cadre de leur domaine respectif puisque les applications de ces travaux touchent, par exemple, les politiques de santé, les décisions au niveau territorial, etc. Dans ce travail, nous produisons, à titre d'exemple, quelques prototypes d'outils d'agrégation et d'exploitation d'informations sémantiques autour des données spatio-temporelles.

Dans le contexte actuel d'accroissement de la production scientifique et avec l'avènement des modèles de publications en accès libre, il devient possible d'accéder à de grands corpus scientifiques pour développer des traitements à grande échelle. Ces types de traitements font appel le plus souvent à de l'apprentissage automatique et, plus récemment, l'apprentissage profond. Cependant, le besoin de proposer des outils capables de fournir des extractions fines et fiables correspondant à des catégories sémantiques dans les textes persiste. L'objectif de cette thèse est d'aborder ces problématiques en proposant des analyses qui s'appuient sur des ressources linguistiques : listes de marqueurs, règles linguistiques d'annotation. Une telle approche par modélisation linguistique permet une généralisation à grande échelle, tout en offrant une traçabilité complète et une fiabilité (mesurable) des résultats [Cardey, 2013]. Ces méthodes peuvent alors servir comme base pour la construction de corpus annotés à grande échelle, nécessaires pour la mise en place des apprentissages.

Pour tester l'approche méthodologique, nous avons choisi de traiter le corpus CORD-19[2], qui est une collection de documents de recherche académique et d'articles scientifiques consacrés à la COVID-19 et aux coronavirus connexes. Le corpus CORD-19 est disponible en accès libre et au format JSON pour les documents en plein texte et au format CSV pour les métadonnées. Ce corpus, dédié à la description et traitement d'une maladie infectieuse, présente une certaine homogénéité disciplinaire. Les publications contiennent une grande quantité d'expressions temporelles, ce qui nous a permis d'observer et d'analyse leur diversité pour la construction des règles d'annotation.

[2] https://github.com/allenai/cord19

Ce travail s'inscrit dans un programme de recherche sous la direction de Dr. Iana Atanassova, mené au laboratoire CRIT (UR 3224) de l'Université de Franche-Comté, autour du développement de méthodes pour la compréhension et la représentation automatique de la sémantique textuelle des corpus scientifiques à grande échelle, de leurs thématiques, structures argumentatives et informations spatio-temporelles. Ce projet fait partie de l'axe de recherche "Sciences, langages, textualités" du CRIT.

Le présent document est organisé de la manière suivante :

1. **Chapitre 1 :** Nous dressons un état de l'art des schémas d'annotation de l'information temporelle, notamment TIMEX2 et TIMEX3, soulignant leur évolution et applicabilité dans la codification des expressions temporelles. Nous discutons également des différentes méthodes d'extraction d'information, mettant en lumière leurs avantages et limites. De plus, nous examinons les outils du domaine de l'extraction et de l'annotation des données temporelles, en parlant de leur efficacité et interopérabilité.

2. **Chapitre 2 :** Dans ce chapitre, nous nous consacrons à l'élaboration d'un corpus d'articles issu de CORD-19, entreprenant une analyse exhaustive de sa structure et de son contenu. De plus, nous mettons en exergue l'importance des ressources de données en libre accès dans le domaine du Traitement Automatique des Langues (TAL).

3. **Chapitre 3 :** Dans ce chapitre, nous abordons TimeInfo, notre schéma conçu pour la catégorisation sémantique des expressions temporelles. TimeInfo est caractérisé par une série d'attributs distincts, chacun associé à des valeurs spécifiques, élaborés en fonction de leur pertinence pour l'annotation de l'information temporelle. Nous examinons en détail chaque attribut de TimeInfo, explorons l'éventail des valeurs applicables et débattons de leur pertinence tant théorique que pratique.

4. **Chapitre 4 :** Dans ce chapitre, nous nous concentrons sur le développement de règles linguistiques spécifiques pour l'annotation sémantique de l'information temporelle selon le schéma TimeInfo. Le chapitre est structuré en deux segments : la première partie se consacre à la conception des règles visant la catégorisation des informations temporelles conformément au schéma TimeInfo. La seconde partie traite de l'implémentation

pratique de ces règles linguistiques, au cours de laquelle nous décrivons l'intégration d'un algorithme destiné à automatiser le processus d'extraction et d'annotation des expressions temporelles.

5. **Chapitre 5 :** Ce chapitre traite de la constitution de TimeTank, un corpus en accès libre qui comprend des phrases enrichies d'expressions temporelles, annotées selon le schéma TimeInfo. La construction de ce corpus a impliqué l'application des règles linguistiques que nous avons développées, ainsi qu'une vérification manuelle pour garantir la précision des annotations.

6. **Chapitre 6 :** Nous entreprenons une analyse comparative détaillée entre TimeInfo et TIMEX3 de TimeML, en nous appuyant sur des exemples concrets extraits des corpus TimeBank et TimeTank. Notre objectif est de mettre en évidence les similitudes et les divergences entre ces deux schémas d'annotation. De plus, nous nous concentrons sur les points où TimeInfo apporte des améliorations ou présente des avantages. Cette analyse vise non seulement à comprendre les nuances entre TimeInfo et TIMEX3 mais aussi à démontrer la valeur ajoutée de TimeInfo dans le contexte de l'annotation sémantique des expressions temporelles.

7. **Chapitre 7 :** Ce chapitre s'oriente vers les applications concrètes et les perspectives de TimeInfo. Nous y explorons l'adaptation ciblée d'un large modèle de langage (LLM) en mettant l'accent sur le fine-tuning en utilisant le corpus TimeTank pour améliorer l'extraction, la catégorisation, et l'annotation de l'information temporelle selon le schéma TimeInfo. Aussi, nous décrivons le développement d'un prototype de moteur de recherche et d'une interface de programmation d'applications (API) qui exploitent le corpus TimeTank.

8. **Chapitre 8 :** Dans ce chapitre, nous abordons la catégorisation sémantique des données spatiales par une méthode à base de règles linguistiques. Nous proposons une expérimentation sur la visualisation des données géospatiales issues du jeu de données CORD-19.

Chapitre 1

Chronologies computationnelles : revue des méthodes d'extraction d'information

Dans un monde où la quantité de données textuelles ne cesse de croître, le besoin d'extraire, d'annoter et de comprendre ces informations est grandissant. Le traitement automatique des langues naturelles s'efforce de relever ces défis à travers diverses méthodes et outils. Ce chapitre se penche en particulier sur l'importance de l'information temporelle au sein des textes, mettant en lumière son rôle central dans la structuration, l'interprétation et l'analyse des contenus. Ainsi, nous étudierons l'évolution des systèmes d'annotation temporelle, depuis les débuts avec TIMEX jusqu'aux normes plus sophistiquées telles qu'ISO-TimeML. Nous explorerons également des corpus annotés tels que TimeBank et les initiatives d'évaluation comme Senseval, SemEval, et TempEval. Le chapitre conclura avec une revue approfondie des outils dédiés à l'extraction et à l'annotation des expressions temporelles, soulignant leurs spécificités, avantages et inconvénients. À travers ce parcours, nous cherchons à offrir une vision globale de la gestion des informations temporelles en traitement automatique des langues.

1.1 L'Annotation et extraction d'information

L'extraction et l'annotation d'informations jouent un rôle central dans le traitement automatique des langues naturelles, en particulier dans le contexte contemporain marqué par une croissance continue des données textuelles. Ces procédés ont pour principal objectif de structurer les données brutes, facilitant ainsi leur interprétation et leur analyse. Cette section vise à étudier les méthodes et techniques d'extraction d'information, de leurs origines basées sur des règles jusqu'aux solutions actuelles exploitant l'apprentissage profond. Elle se penchera également sur les différentes approches d'annotation, soulignant leur pertinence et leur application dans diverses tâches. À travers cette exploration, nous discuterons des avancées, des outils et des défis associés à ces domaines.

1.1.1 Extraction d'information

L'extraction de l'information est une discipline qui se situe à la confluence de la linguistique, de l'informatique et de la statistique [Baeza-Yates et al., 1999]. L'origine de l'extraction d'information peut être retracé à l'analyse des publications scientifiques et des catalogues de bibliothèques [Cowie and Lehnert, 1996]. En effet, c'est ce champ d'expertise qui a jeté les bases des moteurs de recherche modernes que nous employons si couramment aujourd'hui. Dans le contexte des textes scientifiques, cette discipline occupe une position essentielle [Atanassova, 2019]. Les articles, rapports et autres publications regorgent d'informations précises, souvent présentées de manière complexe, rendant leur extraction manuelle fastidieuse et coûteuse [Nasar et al., 2018]. Ainsi, l'extraction automatique de l'information s'inscrit comme une étape cruciale dans le processus d'analyse des textes scientifiques. En effet, avant de pouvoir interpréter, catégoriser ou même visualiser les données, il est essentiel de les extraire de manière structurée. Cette extraction va bien au-delà de la simple reconnaissance de mots-clés. Elle vise à comprendre le contexte, les relations entre les entités et, dans notre projet de recherche, l'importance de l'extraction des données temporelles dans les textes scientifiques.

Avec le temps, de nombreuses méthodes et techniques d'extraction de l'information ont

vu le jour. Ces méthodes peuvent être classées en trois grandes catégories :

1. **Méthodes basées sur les règles :** Les méthodes basées sur les règles ont été parmi les premières utilisées en extraction de l'information. Elles reposent sur la création manuelle de règles et de motifs pour identifier et extraire des informations spécifiques dans les textes [Chiticariu et al., 2013]. Par exemple, une règle pourrait consister à rechercher l'occurrence d'un jour, suivi d'un mois puis d'une année dans un texte afin d'identifier la date.

2. **Approches statistiques :** Ces approches se basent sur l'analyse de la distribution et de la co-occurrence des termes pour identifier des patterns pertinents [Liddy, 2001]. Les modèles comme les Machines à Vecteurs de Support (SVM) ou les modèles de Markov cachés sont des exemples d'outils statistiques utilisés pour l'extraction d'information [Dumais et al., 1998].

3. **Modèles d'apprentissage automatique et profond :** La dernière décennie a vu une explosion de l'utilisation de l'apprentissage profond en extraction de l'information [LeCun et al., 2015]. Ces modèles, tels que les réseaux de neurones récurrents (RNN), les réseaux neuronaux convolutifs (CNN) et plus récemment les architectures Transformer [Wolf et al., 2019], peuvent apprendre de manière autonome des motifs complexes à partir de grandes quantités de données. Grâce à leur capacité à capturer des dépendances entre éléments éloignés et à traiter des contextes variés, ils ont surpassé les performances des méthodes traditionnelles dans de nombreuses tâches d'extraction d'informations.

L'évolution des techniques d'extraction, des méthodes basées sur des règles jusqu'aux modèles d'apprentissage profond, témoigne de la complexité de la tâche d'analyse des textes. Chaque méthode, qu'elle repose sur des règles prédéfinies, des statistiques, ou l'apprentissage machine, joue un rôle dans l'extraction et la catégorisation des données pertinentes.

1.1.2 Annotation des textes

Le processus d'annotation consiste à ajouter des informations structurées à des textes, des images ou des données audio et vidéo. Cette technique est largement utilisée en traitement automatique des langues naturelles, où elle permet de donner une signification précise aux éléments du contenu annoté. L'annotation est essentielle pour de nombreuses tâches de traitement de la langue, telles que la reconnaissance de la parole, la traduction automatique ou l'analyse de sentiments. Elle est également utilisée dans d'autres domaines, tels que la biologie, la médecine ou l'ingénierie, pour extraire et structurer l'information contenue dans des données complexes.

Dans cette partie, nous allons définir en détail ce qu'est l'annotation des textes et expliquer son importance dans le traitement du langage naturel. Nous présenterons également les différents types d'annotations, les outils qui sont utilisés et les étapes du processus d'annotation. Enfin, nous illustrerons ces concepts à l'aide d'exemples concrets tirés de projets de recherche qui ont utilisé des annotations.

Il existe plusieurs approches pour réaliser l'annotation de données. Premièrement, nous avons l'annotation manuelle. Les données annotées manuellement sont celles qui ont été marquées et structurées par un être humain en suivant un schéma d'annotation. L'annotateur humain est souvent un expert dans le domaine concerné. Cette méthode est généralement très précise, mais elle peut être longue et coûteuse à mettre en place si le volume de données à annoter est important.

Dans le cas d'une annotation automatique, les données annotées sont traitées par un programme informatique. Cette approche permet de traiter de grandes quantités de données de manière relativement rapide, mais elle est moins précise que l'annotation manuelle et peut nécessiter un nettoyage et une vérification.

Les approches hybrides combinent l'annotation manuelle et l'annotation automatique. Par exemple, un algorithme peut être utilisé pour pré-traiter les données et proposer des annotations qui seront vérifiées et corrigées par un être humain. Cette méthode permet de combiner l'efficacité de l'annotation automatique avec la précision de l'annotation manuelle.

En ce qui concerne l'annotation des données, qu'elle soit manuelle, automatique ou hybride, elle peut se trouver à l'intérieur du document annoté (In-line annotation) ou à l'extérieur du document annoté (Stand-off annotation).

Dans le premier cas, 'In-line annotation' l'annotation se fait dans le texte lui-même, plutôt que dans un document ou un fichier séparé. Elle est communément utilisée dans le domaine du traitement automatique des langues naturelles et de la linguistique. Par exemple, cette méthode permet aux chercheurs d'annoter le texte avec des informations sur sa structure : titre, auteur, date de publication, paragraphe, etc. Aussi, l'annotation peut s'appliquer aux phrases afin d'identifier les éléments syntaxiques qui la composent : sujet, verbe, complément, etc, ou encore, pour proposer un étiquetage sémantique qui donne des informations sur la signification des expressions et des phrases.

Il existe de nombreuses manières de réaliser un étiquetage en utilisant la méthode 'In-line annotation', dépendant des besoins spécifiques de la tâche d'annotation et des ressources disponibles. 'In-line annotation' est utilisé par les schémas d'annotation de l'information temporelle : TIMEX [Chinchor and Robinson, 1997], TIMEX2 [Ferro et al., 2003], TimeML [Saurí et al., 2006] et le schéma que nous proposons dans la suite de cette thèse TimeInfo.

L'exemple de 'in-line annotation' de la figure 1.1 est extrait de [Pustejovsky et al., 2005] avec le schéma d'annotation TimeML. Annotation de la phrase 'John left 2 days before the attack.' :

```
John
<EVENT eid="e1" class="OCCURRENCE">left</EVENT>
<MAKEINSTANCE eiid="ei1" eventID="e1" tense="PAST" aspect="
   PERFECTIVE"/>
<TIMEX3 tid="t1" type="DURATION" value="P2D"temporalFunction="
   false">2 days</TIMEX3>
<SIGNAL sid="s1">before</SIGNAL>the<EVENT eid="e2" class="
   OCCURRENCE">attack
</EVENT>
<MAKEINSTANCE eiid="ei2" eventID="e2" tense="NONE" aspect="NONE"/
   >.
```

FIGURE 1.1 : Exemple de in-line annotation [Pustejovsky et al., 2005]

L'étiquetage avec la méthode 'stand-off annotation' consiste à stocker les annotations du texte dans un document ou un fichier séparé. Ainsi, le texte original n'est pas modifié.

L'exemple de 'stand-off annotation' ci-après est tiré de l'article [Pustejovsky et al., 2010] où nous avons d'abord une segmentation en tokens 1.2 du texte annoté, suivie d'une deuxième partie1.3 où les éléments sont annotés en utilisant le schéma ISO-TimeML. Ci-dessous nous avon un exe Tokenisation

```
<maf xmlns:"http://www.iso.org/maf">
    <seg type="token" xml:id="token1">Mia</seg>
    <seg type="token" xml:id="token2">visited</seg>
    <seg type="token" xml:id="token3">Seoul</seg>
    <seg type="token" xml:id="token4">to</seg>
    <seg type="token" xml:id="token5">look</seg>
    <seg type="token" xml:id="token6">me</seg>
    <seg type="token" xml:id="token7">up</seg>
    <seg type="token" xml:id="token8">yesterday</seg>
    <pc>.</pc>
</maf}
```

FIGURE 1.2 : Exemple de stand-off annotation : tokenisation

```
<isoTimeML
    xmlns:"http://www.iso.org./isoTimeML">
    <TIMEX3 xml:id="t0" type="DATE"
    value="2009-10-20" functionInDocument="CREATION_TIME"/>
    <EVENT xml:id="e1" target="#token2" class="OCCURRENCE" tense=
        "PAST"/>
    <EVENT xml:id="e2" target="#range(#token5,#token7)" class="
        OCCURRENCE"
    tense="NONE" vForm="INFINITIVE"/>
    <TIMEX3 xml:id="t1" type="DATE" value="2009-10-19"/>
    <TLINK target="#range(#e1,#t0)" relType="BEFORE"/>
    <TLINK target="#range(#e1,#t1)" relType="ON_OR_BEFORE"/>
    <TLINK target="#range(#e2,#t1)" relType="IS_INCLUDED"/>
</isoTimeML}
```

FIGURE 1.3 : Exemple de stand-off annotation : annotation

Browser-based Annotation Tool ou Brat [Stenetorp et al., 2012b] est un outil d'annotation de données en traitement du langage naturel. Il permet à un utilisateur de marquer et de struc-

turer du texte ou des documents en ajoutant des étiquettes. Brat est facile à utiliser, il peut être lancé dans un navigateur web [Stenetorp et al., 2012a]. Brat a été utilisé dans de nombreux projets de recherche, notamment pour l'annotation du discours [Mehrabani et al., 2015], l'annotation des événements [Araki et al., 2018], de journaux médicaux [Huang and Lu, 2016], etc.

Comme Brat, WebAnno [Yimam et al., 2013] est un outil d'annotation de données qui permet à un utilisateur de marquer et de structurer du texte ou des documents en ajoutant des étiquettes [De Castilho et al., 2016]. WebAnno offre de nombreuses fonctionnalités avancées pour l'annotation, telles que la prise en charge de l'annotation à plusieurs niveaux, la possibilité de définir des règles de validation des annotations et la possibilité de travailler en mode collaboratif avec plusieurs utilisateurs. WebAnno est également compatible avec de nombreux formats de données et peut être intégré à d'autres outils de traitement du langage naturel [Yimam et al., 2014]. WebAnno a été utilisé dans de nombreux projets de recherches, notamment dans le traitement automatique des langues naturelles [Yimam, 2019], dans le biomédical [Yimam et al., 2015], dans l'économique et la finance [Jacobs and Hoste, 2022], etc.

Dans notre projet de recherche, nous n'utilisons pas les outils existants tels Brat et WebAnno. En effet, nous trouvons avantageux de développer nos propres outils d'annotation et cela pour plusieurs raisons. Tout d'abord, cela permet d'avoir plus de liberté et de flexibilité dans le processus d'annotation. En développant ses propres outils, il est possible de personnaliser les fonctionnalités et les règles d'annotation pour répondre aux besoins spécifiques de notre projet de recherche. Ainsi, nous pouvons nous baser sur notre schéma d'annotation de l'information temporelle pour l'extraction et l'annotation sémantique des données.

Ensuite, le développement de ses propres outils permet d'obtenir une précision supérieure. En construisant un outil sur mesure, il est possible de s'assurer que les règles d'annotation sont bien adaptées à notre corpus et qu'il n'y a pas de biais ou d'erreurs introduits par l'utilisation d'un outil existant.

Il est manifeste que l'extraction et l'annotation d'informations constituent des pierres angulaires du traitement automatique des langues naturelles. Des origines basées sur des règles

jusqu'aux méthodes sophistiquées d'apprentissage profond, l'évolution des techniques d'extraction illustre la nécessité croissante de traiter et d'interpréter d'énormes volumes de données textuelles de manière efficace. L'annotation, qu'elle soit réalisée manuellement, automatiquement ou via des approches hybrides, sert à enrichir ces données.

1.2 Les méthodes basées sur les corpus

Les méthodes basées sur les corpus nous aident à explorer la langue dans son contexte naturel. En effet, nous n'avons pas besoin de générer de nouvelles données, mais d'utiliser un corpus déjà existant.

L'un des objectifs du Traitement Automatique des Langues est d'attribuer une annotation riche à des simples chaînes linéaires de mots. [Brill, 1993] Avant d'arriver à l'annotation, il faut construire une base de connaissances qui est utilisée par l'algorithme afin d'annoter le texte d'entrée. Cette base de connaissances est acquise grâce l'analyse et l'étude d'un corpus donné. L'un des nombreux avantages qu'offrent les méthodes à bases de corpus est de nous permettre une analyse empirique des phénomènes linguistiques présents dans le corpus étudié [Biber et al., 1994].

1.2.1 Analyse sémantique

Dans le domaine de l'interprétation sémantique, il y a eu un certain nombre d'utilisations intéressantes des méthodes basées sur les corpus. Ces méthodes sont utilisées afin de développer des règles pour sélectionner le sens approprié d'un mot sémantiquement ambigu [Ng and Zelle, 1997].

Pour l'analyse sémantique, [Gries and Divjak, 2009] soutiennent que les méthodes basées sur les corpus, telles que 'Near-synonymy' (voir [Hirst, 1995] et [Edmonds and Hirst, 2002]) et 'Polysemy' (voir le chapitre 10 de [Ravin and Leacock, 2000]) présentent de nombreux avantages par rapport aux autres approches. En effet, les méthodes basées sur les corpus présentent un certain nombre d'avantages, notamment, la possibilité d'analyser les données dans

leurs contextes naturels, l'utilisation des méthodes d'apprentissage empiriques et application des méthodes de traitement automatique des langues et le fait que les résultats ne peuvent pas être influencés par des connaissances implicites.

1.2.2 Signaux temporels

[Derczynski and Gaizauskas, 2012] examinent comment les signaux temporels sont utilisés dans le corpus TimeBank afin de déterminer les relations temporelles entre les événements. Ils expliquent comment les signaux temporelles tels que 'after, when, until, previously, during', etc, montrent l'existence d'une relation temporelle entre deux événements donnés. Cette étude a révélé qu'une grande partie des relations temporelles sont signalées par des expressions temporelles et que ces expressions sont souvent ambiguës. Par ailleurs, [Velupillai et al., 2015] décrit un système d'extraction d'informations temporelles à partir de données cliniques. Le système utilisé dans cette étude se base sur des expressions régulières et des approches d'apprentissage automatique. L'annotation utilisée est basée sur le schéma TimeML [Pustejovsky et al., 2005].

1.3 Systèmes, architectures et directives pour l'annotation des informations temporelles

Certains repères temporels sont explicitement ancrés à la phrase, comme dans l'exemple :'The mortality of the 27 included patients infected by 2019-nCoV was 37%, which is much higher than that reported 2% on **4 Feb 2020**.' [Yuan et al., 2020]. D'autres nécessitent une connaissance contextuelle pour être correctement interprétés, comme nous pouvons le voir dans : 'However, it was not until **last year** that the anti-influenza virus effects of chloroquine at clinically achievable concentrations were studied, in view of a possible application of this drug in the clinical management of influenza [4,15].'[Di Trani et al., 2007].

Il convient de mentionner l'existence de systèmes d'annotation capables de traiter à la fois les informations temporelles absolues et relatives. Les sections suivantes se pencheront

sur ces systèmes, abordant leur utilité, des cas d'utilisation spécifiques, ainsi que leurs potentialités et contraintes.

1.3.1 Système d'annotation de l'information Temporelle : TIMEX

Dans cette partie nous nous intéressons au système d'annotation TIMEX, l'un des premiers dédiés à l'annotation de l'information temporelle. TIMEX a été introduit lors de la conférence MUC-7, parrainée par la 'Defense Advanced Research Projects Agency' (DARPA) [Chinchor, 1998]. L'annotation temporelle est souvent considérée comme une spécialisation de la reconnaissance d'entités nommées. Dans le cadre du MUC-7, cette annotation se réalise via la balise TIMEX [Chinchor and Robinson, 1997]. Notons que cette balise TIMEX est alignée avec le langage de balisage 'Standard Generalized Markup Language' (SGML) [Goldfarb, 1985].

Approfondissons notre compréhension de la balise TIMEX telle qu'elle est présentée dans MUC-7 [Chinchor and Robinson, 1997]. TIMEX comporte un attribut essentiel : TYPE. Cet attribut peut prendre les valeurs DATE ou TIME. DATE se réfère à une date qui est représentée, totalement ou partiellement, dans un texte, tandis que TIME renvoie à une période de la journée, comme les heures ou les minutes. En synthèse, DATE correspond à une durée supérieure à 24 heures et TIME à une durée inférieure à 24 heures.

La conférence MUC-7 spécifie également plusieurs types d'expressions temporelles, notamment les expressions temporelles absolues et relatives. Une expression est jugée absolue si elle est explicite. Des exemples typiques incluent les jours de la semaine (lundi, mardi) ou les décennies (1980s, 1990s). Ainsi, tout repère temporel précis, qu'il concerne une heure, un jour, un mois, une saison, une année, est considéré comme une information temporelle absolue.

Inversement, l'information temporelle relative ne fournit pas une valeur temporelle exacte, mais seulement une indication partielle de l'unité temporelle en contexte [Chinchor and Robinson, 1997]. Prenons les exemples suivants pour clarifier la différence :

1. 'Numerous malaria epidemics have occurred in western Kenya, with increasing fre-

quency over the past 20 years.' [Carlson et al., 2004]

2. 'For example, it is possible that the population was exposed to canine parvovirus in the 1980s [11].' [Ellis and Post, 2004]

Dans le premier cas, 'over the past 20 years' est une expression relative puisque sa complétude dépend de la date d'énonciation. TimeML propose une solution pour déterminer cette date en extrayant l'information directement du document source grâce à l'attribut 'functionInDocument' [Saurí et al., 2006]. Cependant, cette méthode n'est pas toujours fiable, notamment pour les pages web, où la date peut être celle de la dernière consultation ou mise à jour. Dans notre recherche, nous priorisons l'information temporelle absolue des articles scientifiques. Toutefois, la date de publication peut fournir un repère utile pour estimer la date d'énonciation d'une information temporelle.

Quant au deuxième exemple, 'in the 1980s', c'est clairement une expression temporelle absolue car elle se réfère à une décennie précise, identifiable hors contexte.

Bien que MUC-7 présente à la fois les informations temporelles absolues et relatives, le système d'annotation TIMEX ne fait pas cette distinction. Voici comment les exemples précédents sont annotés avec TIMEX :

<TIMEX TYPE= "DATE">over the past 20 years<TIMEX>

<TIMEX TYPE= "DATE"> in the 1980s<TIMEX>

1.3.2 TIMEX2

La présente section se consacre à la première mise à jour significative du système d'annotation temporelle, TIMEX, évoqué précédemment. Nous aborderons ici le système TIMEX2, une innovation résultant du programme de recherche TIDES [Ferro et al., 2003].

TIMEX2 a été conçu dans le dessein d'établir un ensemble cohérent de directives pour l'annotation et la normalisation de l'information temporelle. Cette normalisation renvoie à la conversion d'informations temporelles exprimées en langage courant vers une forme conforme à un standard international, à savoir ISO 8601[1]. Bien que la référence à ce standard apporte une structure précise au système d'annotation, des limitations demeurent, principalement dues aux contraintes imposées par le traitement du langage naturel. Pour pallier ces limitations, les chercheurs impliqués dans le projet TIDES ont enrichi le standard ISO avec des unités temporelles nouvelles, telles que les siècles ou les décennies, absentes de l'ISO 8601. Ainsi, ils ont introduit des unités telles que 'DE' pour 'Decade' et 'CE' pour 'Century' (comme détaillé dans la section 4.2.4.2 de TIDES [Ferro et al., 2003]).

Le projet TIDES a souligné deux principales applications pour le schéma d'annotation TIMEX2 [Ferro et al., 2003]. La première vise à servir de guide aux annotateurs humains pour standardiser l'annotation de l'information temporelle. L'autre est conçue pour faciliter la création de logiciels dédiés à l'extraction et à l'annotation automatisée de l'information temporelle, comme illustré dans [Mani et al., 2001]. Un aperçu détaillé de la structure de TIMEX2 est présenté ci-dessous (voir Figure 1.4).

Le projet de recherche TIDES définit qu'une information temporelle est étiquetable lorsqu'elle contient un élément lexical déclencheur, tel que des noms, adjectifs ou adverbes, comme le détaille la section 3.1 de [Ferro et al., 2003]. Nous abordons ici les attributs essentiels du système d'annotation temporelle TIMEX2.

L'attribut 'VAL' est fondamental, car il identifie la valeur normalisée de l'information temporelle. L'attribut 'ANCHOR_VAL', quant à lui, exprime une durée et est systématiquement associé à l'attribut 'ANCHOR_DIR'. L'attribut 'ANCHOR_VAL' représente la valeur

[1] https://www.iso.org/fr/iso-8601-date-and-time-format.html

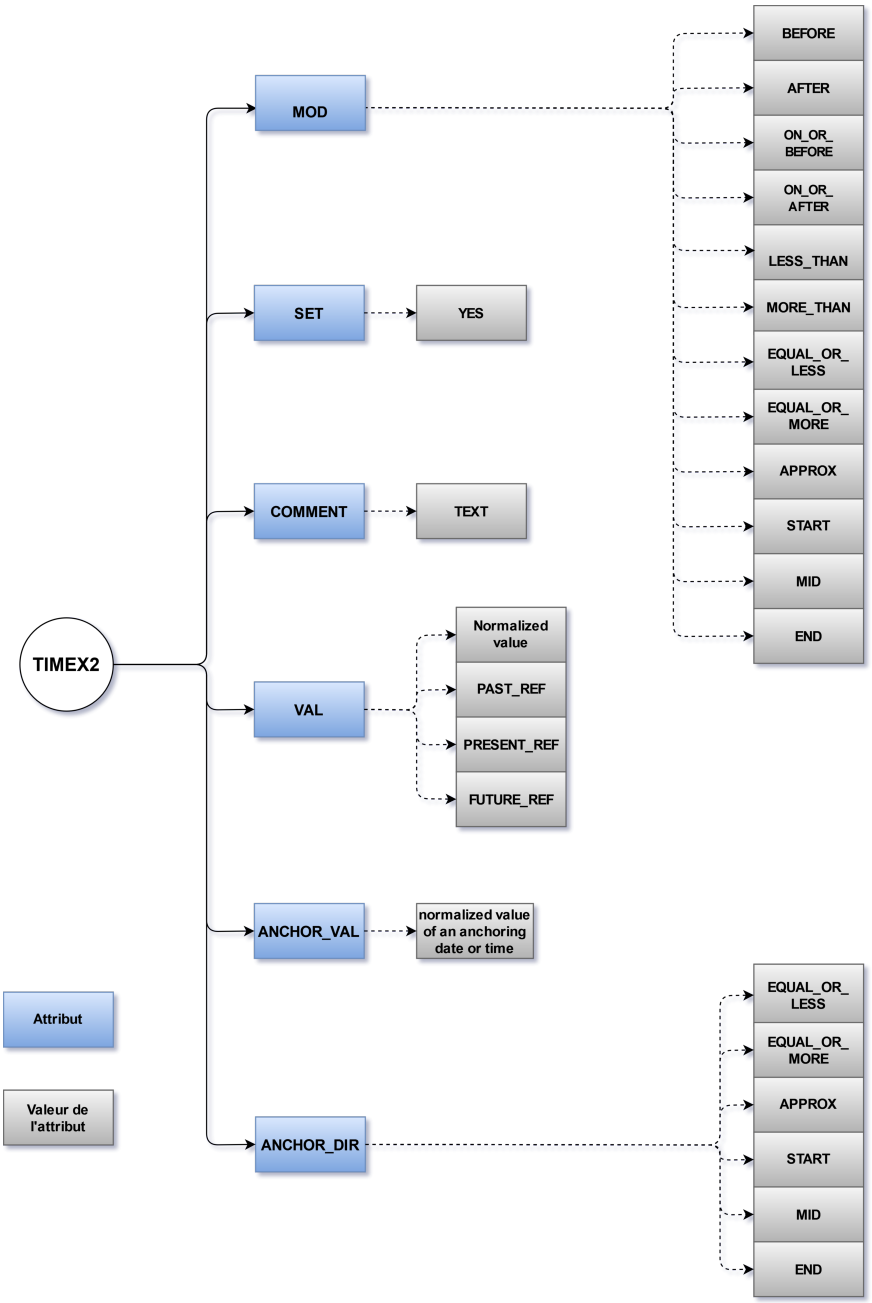


FIGURE 1.4 : Système d'annotation TIMEX2

temporelle normalisée selon l'ISO 8601, tandis que 'ANCHOR_DIR' fournit une orientation ou direction entre les informations de 'VAL' et 'ANCHOR_VAL'. Certaines des valeurs possibles pour 'ANCHOR_DIR' incluent : 'starting', 'ending', 'before', entre autres (voir section 4.2.4 de [Ferro et al., 2003]).

L'attribut final de TIMEX2 à aborder est 'SET', spécifiquement conçu pour l'information temporelle récurrente, identifiée à l'aide d'adverbes tels que 'every' ou des formes plurielles des éléments déclencheurs mentionnés précédemment (cf. section 3.1 de [Ferro et al., 2003]). Lorsqu'une récurrence est identifiée, 'SET' est assigné à la valeur 'YES'. Par exemple : 'They watched Millionaire on TV <TIMEX2 SET="YES" VAL="1999-WXX-2"> every Tuesday in <TIMEX2 VAL="1999">1999</TIMEX2></TIMEX2>.' (pour plus d'exemples, voir la section 4.5.1 de [Ferro et al., 2003]).

1.3.3 TIMEX3 et TimeML

TimeML est un langage d'annotation conçu pour la reconnaissance et l'ancrage des événements et des expressions temporelles, ainsi que des relations qu'ils entretiennent. Il a été élaboré lors de l'atelier TERQAS [Saurí et al., 2006].

Initialement introduit au début des années 2000, TimeML est né d'un besoin croissant de standardiser la manière dont les informations temporelles sont représentées et traitées dans les textes. Au fil des années, TimeML a été amélioré et adapté pour répondre aux besoins changeants des chercheurs et des professionnels, menant à l'introduction de diverses balises et structures, dont TIMEX3 [Pustejovsky et al., 2003a].

Le système d'annotation TimeML est détaillé, avec des balises telles que <EVENT> pour les événements, <TIMEX3> pour les expressions temporelles, et <SIGNAL> pour les relations entre les différentes balises mentionnées précédemment [Saurí et al., 2006], la figure 1.5 illustre ces relations entres les différentes balises.

Notre étude se concentre principalement sur l'information temporelle, représentée par la balise <TIMEX3>, qui a été inspirée et enrichie à partir des attributs de TIMEX2 du projet TIDES, discuté dans la section précédente.

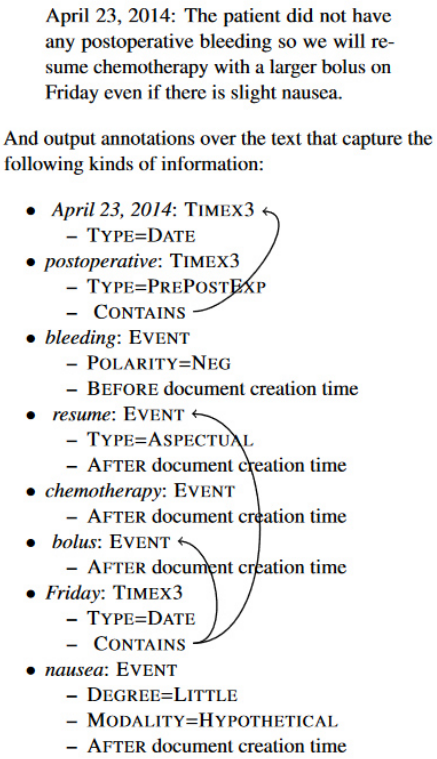


FIGURE 1.5 : Exemple utilisant TimeML, extrait de [Bethard et al., 2015]

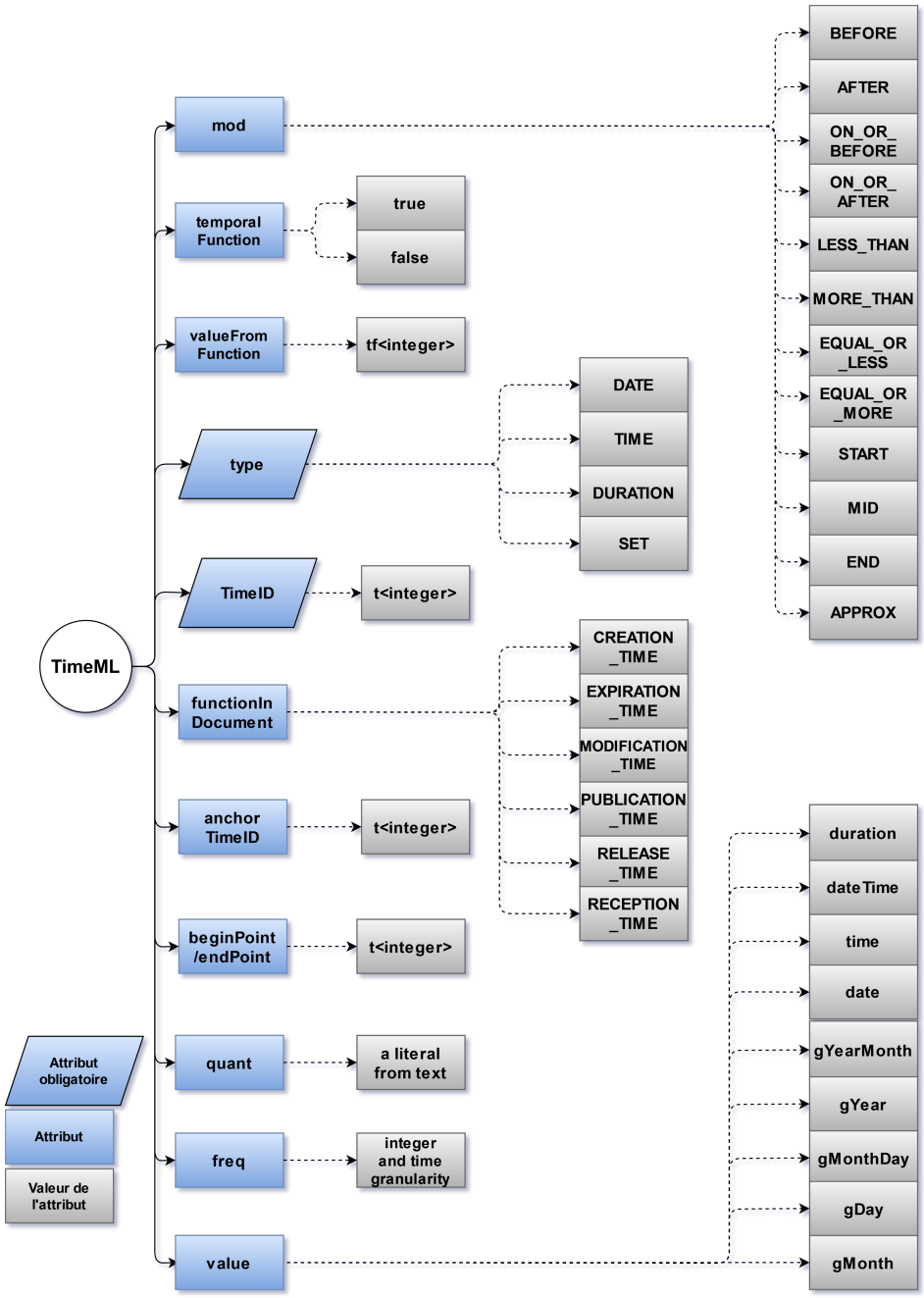


FIGURE 1.6 : Système d'annotation TIMEX3

La Figure 1.6 détaille le schéma d'annotation de l'information temporelle de TimeML. La balise TIMEX3 possède dix attributs, certains ayant une valeur unique, comme TimeID et valueFromFunction, tandis que d'autres peuvent prendre plusieurs valeurs, comme les attributs TYPE et VALUE.

Nous discuterons maintenant en détail des attributs principaux de TIMEX3.

L'attribut TYPE catégorise l'information temporelle. Ses valeurs possibles sont : DATE, TIME, DURATION et SET. La catégorie DATE englobe toutes les informations temporelles pouvant être représentées dans un calendrier [Saurí et al., 2006], tandis que TIME se réfère à un moment spécifique de la journée. Ces deux catégories sont similaires à celles définies dans MUC-7. L'attribut DURATION représente une durée, mais si cette durée peut être fixée à une date ou une heure précise, elle est alors catégorisée respectivement en DATE ou TIME. Pour plus de détails, consulter [Saurí et al., 2006].

Enfin, SET, la dernière valeur possible pour l'attribut TYPE, traite des expressions temporelles récurrentes, semblable à la définition donnée dans TIMEX2 de TIDES [Ferro et al., 2003].

Les principaux **avantages** de TimeML sont sa granularité et sa flexibilité. Il offre des balises spécifiques pour divers types d'informations temporelles, permettant une annotation détaillée des textes. De plus, étant un standard, il favorise la collaboration et la comparaison des travaux entre chercheurs. Cependant, TimeML présente certains **inconvénients**. Sa complexité peut être un défi pour les nouveaux utilisateurs ou ceux sans une formation approfondie. L'annotation manuelle selon les normes TimeML peut être longue et exigeante. De plus, bien que TimeML soit adaptatif, ses structures sont trop rigides ou ne capturent pas toutes les nuances des informations temporelles, en particulier les expressions temporelles complexes.

Il est à noter que TIMEX3, bien que s'inspirant de TIMEX2, ajoute des liaisons entre les événements et les expressions temporelles. Tant TIMEX2 que TIMEX3 peuvent servir de référence pour l'annotation manuelle ou la conception de logiciels destinés à l'extraction automatique d'informations temporelles, à l'instar de HeidelTime[2].

[2] https://heideltime.ifi.uni-heidelberg.de/heideltime/

1.3.4 ISO-TimeML

La naissance d'ISO-TimeML[3] découle d'un besoin croissant de mieux traiter et représenter les informations temporelles. TimeML, bien qu'innovant présente certaines limitations, en particulier en ce qui concerne l'interopérabilité avec d'autres standards et l'efficacité de l'annotation pour des corpus de grande taille. L'évolution vers ISO-TimeML a ainsi permis une standardisation accrue et une approche d'annotation plus sophistiquée.

Ainsi, ISO-TimeML est perçu comme une évolution de TimeML, visant une meilleure interopérabilité. Ce dernier propose des modifications ainsi que des enrichissements concernant les informations temporelles. Cette mise à jour s'appuie sur une méthodologie d'annotation sémantique qui établit une distinction fondamentale entre l'annotation d'une expression et la représentation que cette dernière dénote [Pustejovsky et al., 2010]. De plus, ISO-TimeML a été élaboré conformément aux standards de The International Organization for Standardization (ISO).

À l'instar de TimeML, ISO-TimeML emploie des balises et des attributs pour annoter les textes contenant des données temporelles, comme les événements, expressions et relations temporelles. Ce standard a été largement adopté par la communauté de traitement automatique des langues naturelles et a servi pour diverses tâches, à savoir :

- **Extraction d'événements** : L'annotation permet d'obtenir des informations sur les événements, tels que leur occurrence, les acteurs impliqués et leur nature.
- **Extraction de relations temporelles** : Des balises spécifiques sont dédiées à représenter les relations temporelles entre les événements et expressions, comme "avant", "après" ou "simultané".
- **Synthèse** : Les annotations révèlent les principaux événements et expressions temporelles d'un texte, facilitant ainsi sa synthèse.
- **Extraction d'informations** : Outre les événements, les annotations renseignent sur les expressions et relations temporelles. Ces données sont cruciales pour alimen-

[3] https://www.iso.org/fr/standard/37331.html

ter des bases de données ou constituer des corpus, comme le TimeBank Dataset [Pustejovsky et al., 2003b].

- **Traduction automatique** : Les annotations dans une langue peuvent faciliter la traduction du texte dans une autre langue tout en conservant les nuances temporelles.

L'une des transformations majeures de TimeML vers ISO-TimeML réside dans le type d'annotation. Contrairement à TimeML qui utilise une 'in-line annotation', ISO-TimeML privilégie la 'Standoff Annotation' [Pustejovsky et al., 2010]. Pour plus de détails sur ces méthodes d'annotation, consulter la section 1.3.

De nombreux projets ont intégré ISO-TimeML, notamment :

- **TimeBank** : Un corpus largement cité qui utilise ISO-TimeML pour annoter les événements et relations temporelles [Pustejovsky et al., 2003b].
- **TempEval** : Une série de compétitions d'évaluation ayant pour but d'identifier et d'analyser les informations temporelles dans les textes selon les standards d'ISO-TimeML [Verhagen et al., 2007].
- **TERSEO** : Un projet destiné à l'annotation semi-automatique des événements et des expressions temporelles dans les textes en espagnol [Saquete et al., 2006].
- **NewsReader** : Un projet qui vise à extraire des informations à partir de nouvelles en utilisant, entre autres, ISO-TimeML pour la gestion des informations temporelles [Vossen et al., 2016].

Malgré ses avantages, ISO-TimeML présente également des limites. Ces dernières concernent notamment sa complexité et sa difficulté de prise en main, ainsi que son incapacité à prendre en compte les expressions temporelles complexes.

1.4 TimeBank

Le corpus TimeBank est un ensemble structuré de 183 articles de presse, minutieusement annotés, issu de l'effort collaboratif de l'Université Brandeis. Ce corpus est hébergé par le Linguistic Data Consortium (LDC)[4]. L'annotation dans TimeBank se concentre sur trois principaux éléments : les expressions temporelles, les événements, et les relations temporelles entre ces éléments. Les annotations sont effectuées selon le schéma TimeML [Setzer, 2002].

La conception de TimeBank était motivée par une quête de réponses à des questions cruciales concernant la dimension temporelle, notamment, comment les événements sont-ils positionnés et ordonnés chronologiquement [Pustejovsky et al., 2003b]. L'importance de TimeBank ne se limite pas à sa structure, mais réside également dans sa fonction en tant que ressource pour la recherche. Il sert de fondement pour le développement d'algorithmes et d'outils visant à reconnaître les expressions temporelles, et pour effectuer une annotation automatique conforme au schéma TimeML. En outre, TimeBank joue un rôle pivot dans l'entraînement et l'évaluation de modèles axés sur l'extraction d'informations temporelles adaptés à une panoplie d'applications. Il est aussi utilisé spécifiquement pour des tâches telles que la détection d'événements et la résolution de référence d'événements. Ci-dessous un exemple extrait du corpus TimeBank :

[4] https://catalog.ldc.upenn.edu/LDC2006T08

```
Police <EVENT eid="e8" class="REPORTING">confirmed</EVENT>
<TIMEX3 tid="t39" type="DATE" value="1998-02-13" temporalFunction="
   true" functionInDocument="NONE" anchorTimeID="t30">Friday</
   TIMEX3>that the body
 <EVENT eid="e9"class="OCCURRENCE">found</EVENT> along a highway in
     this municipality 15 miles south of San Juan
 <EVENT eid="e10" class="STATE">belonged</EVENT>  to Jorge
    Hernandez, 49. Hernandez  was
 <EVENT  eid="e11" class="OCCURRENCE">kidnapped</EVENT> from his
    small, neighborhood store in the town of Trujillo Alto
<SIGNAL sid="s45">at</SIGNAL>
<TIMEX3 tid="t35" type="TIME"value="1998-02-11T22:00"
   temporalFunction="true" functionInDocument="NONE" anchorTimeID="
   t30">10 p.m. Wednesday</TIMEX3>, police
<EVENT eid="e12" class="REPORTING">said</EVENT>. His kidnappers
<EVENT eid="e13" class="I_ACTION">demanded</EVENT> a
<EVENT eid="e162" class="OCCURRENCE">ransom</EVENT> of dlrs 1
   million, but
<EVENT eid="e47"class="OCCURRENCE">negotiations</EVENT>
<EVENT eid="e14" class="ASPECTUAL">broke</EVENT> off
<SIGNAL sid="s187">at</SIGNAL>
<TIMEX3 tid="t36" type="TIME" value="1998-02-12T12:00"
   temporalFunction="true" functionInDocument="NONE" anchorTimeID="
   t30">noon Thursday </TIMEX3>. Police
<EVENT eid="e15" class="REPORTING">gave</EVENT> no details about
   the
<EVENT  eid="e48" class="I_ACTION">negotiations</EVENT> with the
   kidnappers for the
<EVENT eid="e50"class="OCCURRENCE">return</EVENT> of Hernandez.
```

Le corpus TimeBank est un outil précieux pour la recherche en analyse des données tem-

porelle. Cependant, il présente également certaines limites qui doivent être prises en compte. D'une part, la taille du corpus est relativement petit, ce qui peut limiter la généralisabilité des modèles formés sur ce corpus à d'autres contextes ou applications. D'autre part, une grande partie des données temporelles annotées sont des expressions temporelles relatives.

Dans le chapitre 6, nous nous penchons de manière approfondie sur le corpus TimeBank. Ce dernier sert de référence pour la mise en comparaison de TIMEX3, le standard d'annotation des expressions temporelles dans TimeML et ISO-TimeML, avec notre propre schéma d'annotation de l'information temporelle.

1.5 SensEval, SemEval et TempEval

L'efficacité des innovations dépend largement de la rigueur avec laquelle elles sont évaluées. La mise en place de cadres d'évaluation solides est impérative, car elle sert de pierre angulaire à l'avancement de la recherche, établissant des repères, catalysant la collaboration, et instaurant un niveau de standardisation. Au cœur de cette dynamique se trouvent des initiatives comme Senseval, SemEval, et TempEval.

Senseval, initié en 1998 en Angleterre [Edmonds and Cotton, 2001], fut une série d'évaluations conçues pour jauger les méthodes de traitement computationnel dans diverses tâches, telles que la désambiguïsation sémantique [Edmonds, 2002] et l'échantillonnage lexical [Mihalcea et al., 2004]. Ces évaluations ont considérablement influencé le progrès en traitement automatique des langues[5].

Suite à *Senseval*, *SemEval*[6] a émergé comme une plateforme d'évaluation multidimensionnelle pour le traitement du langage naturel. Outre les tâches traditionnelles telles que la classification de texte, SemEval introduit des défis liés à l'analyse de sentiments ou encore la reconnaissance d'entités. La synergie créée lors de ces ateliers et l'accent mis sur l'open source ont permis un partage prolifique de techniques et d'approches entre participants.

TempEval, en revanche, se concentre exclusivement sur l'évaluation des systèmes

[5] https://web.eecs.umich.edu/~mihalcea/senseval/past.html
[6] https://semeval.github.io/

annotant les relations temporelles, ancré dans le schéma d'annotation TimeML. Lancé lors de SemEval 2007 [Verhagen et al., 2007], ce cadre se divise en trois sous-tâches précises, chacune dédiée à l'annotation d'un aspect spécifique de l'information temporelle [Verhagen et al., 2009]. Les annotations sont effectuées au moyen de balises telles que *TempEval*, `s`, `TIMEX3`, `EVENT` et `TLINK`.

Avec le temps, *TempEval* s'est adapté pour évaluer les systèmes traitant divers types de textes, tels que les articles de presse, les tweets ou les documents biomédicaux. De plus, il s'est internationalisé pour englober plusieurs langues, français, chinois, etc [Llorens et al., 2011]. Toutefois, malgré ses contributions significatives, *TempEval* présente des défis inhérents à sa dépendance à l'architecture TimeML. Les expressions temporelles complexes et les nuances linguistiques spécifiques peuvent parfois entraver son efficacité.

La Figure 1.7 offre une synthèse visuelle des distinctions de SensEval, SemEval et TempEva, mettant en lumière la complémentarité et la progression de ces initiatives.

Bien que *TempEval* ait facilité l'évaluation de nombreux systèmes de traitement d'information temporelle, ses limitations, notamment celles liées à la structure de TimeML, nous ont amenés à envisager d'autres métriques, comme la précision, le rappel et le F-mesure, pour évaluer des outils développés à partir de notre méthodologie de l'annotation de l'information temporelle.

FIGURE 1.7 : Différences et évolutions entre Senseval, SemEval, et TempEval

1.6 Gestion et annotation des informations temporelles en TAL

La gestion, l'annotation et la compréhension des informations temporelles sont des pierres angulaires du Traitement Automatique des Langues. Dans le contexte des textes d'actualité et, plus particulièrement, de l'actualité financière, l'identification et la liaison des entités temporelles avec des événements précis sont cruciales. Cela nécessite une approche multidimensionnelle qui englobe non seulement la détection de ces entités, mais également la compréhension de leur contexte, leur hiérarchisation et leur relation avec d'autres éléments du texte. Ici, nous mettons en lumière des travaux de recherche majeurs qui abordent ces dimensions, en offrant des aperçus sur la représentation de la connaissance temporelle, le raisonnement temporel et la catégorisation sémantique de l'information temporelle.

[Schilder and Habel, 2001] décrivent un système d'annotation sémantique qui extrait les informations temporelles des textes qui traitent de l'actualité. L'annotation sémantique se focalise sur les dates et les phrases prépositionnelles contenant des expressions temporelles. Le système utilise des transducteurs finis basés sur des règles entraînées sur des articles d'actualité économique. Le système propose une représentation sémantique des expressions temporelles extraites, et vise à établir des relations temporelles avec les événements mentionnés dans un article de presse. Dans ce projet de recherche, les auteurs font une distinction entre les informations temporelles et les expressions qui font référence à des événements. Une fois la distinction faite, une fonction est créée afin de lier l'information temporelle à l'événement.

Le système présenté par [Schilder and Habel, 2001] est un système de balisage ou d'étiquetage sémantique de l'information temporelle et des événements axé sur l'actualité financière. Pour ce faire, un ensemble de règles finies est utilisé pour permettre l'extraction et l'étiquetage de l'information temporelle et des événements. Néanmoins, la méthodologie de l'annotation sémantique proposée par [Schilder and Habel, 2001] se résume à lier un événement à une information temporelle et à des relations temporelles qui sont explicitement marquées par des prépositions temporelles (pour des exemples, voir [Schilder and Habel, 2001] section 2.2.2). Ces relations temporelles sont celles définies par [Allen, 1983], l'étude in-

troduit une représentation temporelle basée sur des intervalles qui utilise des techniques de propagation de contraintes et des intervalles de référence pour représenter les relations entre des intervalles temporels de manière hiérarchique. Ainsi, l'étude de [Allen, 1983] soutient que des intervalles temporels devraient être utilisés comme primitifs plutôt que des points temporels, car ils nous permettent de capturer la hiérarchie des relations temporelles entre les événements ainsi que de faciliter les processus de raisonnement. Les relations proposées dans cette études sont exprimées de la manière suivante : X before Y, X meets Y, X overlaps Y, X during Y, etc. Pour conclure avec l'étude [Schilder and Habel, 2001] et [Allen, 1983], nous avons des relations temporelles basées sur l'utilisation d'une hiérarchie d'intervalles de référence. Cette approche peut être utile dans des domaines où les informations temporelles sont imprécises et relatives.

Dans l'étude de [Mani et al., 2006], les chercheurs ont exploré l'annotation des relations temporelles en utilisant le langage de balisage TimeML. En combinant avec des méthodes d'apprentissage automatique, ils ont cherché à déduire les relations temporelles entre les événements et les temps dans des articles de presse. Plusieurs méthodes de référence, dont des règles codées à la main et des règles hybrides incorporant des données de Google (VerbOcean) [Chklovski and Pantel, 2004], ont été évaluées. Les résultats montrent clairement que l'approche basée sur l'apprentissage automatique surpassait les méthodes basées sur l'intuition. Cependant, l'étude a rencontré plusieurs défis, notamment la rareté des données. La méthode utilisée ne tient pas compte des dépendances potentielles entre les paires de relations. De plus, les règles basées sur l'intuition, même lorsqu'elles sont enrichies avec des données comme VerbOcean, ont une portée limitée. En outre, le corpus utilisé présente un certain niveau de bruit, en particulier le TimeBank. Ces résultats soulignent l'importance de l'intégration de la logique temporelle avec l'apprentissage automatique, tout en reconnaissant les défis associés à l'extraction de relations à partir de données textuelles.

1.7 Travaux autour de l'extraction d'informations menés au Centre Tesnière

Notre travail dans cette thèse prend comme fondement méthodologique les travaux sur l'annotation sémantique et l'extraction d'informations menés au Centre Tesnière (laboratoire CRIT, Université de Franche-Comté). En particulier :

- L'approche microsystémique de la langue, introduite par S. Cardey [Cardey, 2013] qui propose une méthodologie pour analyser la langue en tant que système composé de micro-systèmes, permettant la mise en place d'analyses automatisées.
- Les travaux autour de l'extraction d'informations : S. Cardey [Cardey, 2022] propose une représentation formelle des règles de reconnaissance, appliquée aux données personnelles dans les textes. [Atanassova et al., 2021] et [Atanassova et al., 2022] introduisent une méthodologie pour l'identification et la gestion des données personnelles dans les textes. [Guisse and Atanassova, 2022] propose une implémentation informatique des règles linguistiques pour le traitement des données personnelles. Plus largement, le numéro 40 de la revue Bulag [Cardey et al., 2022], dédié à Peter Greenfield, présente un grand nombre de travaux autour des analyses linguistiques automatisables.

Une série de projets ont été développés au Centre Tesnière, sous la direction de Sylviane Cardey, en collaboration avec des entreprises, pour produire des outils d'analyse de corpus textuels pour différentes applications.

Le projet Interreg DecRIPT[7] (Détection des diverses Représentations de l'Information permettant d'identifier les données Personnelles contenues dans les Textes), en collaboration avec la HEG de Genève, la Haute école de gestion Arc, et les entreprises ERDIL et Global Data Excellence, a permis de proposer un modèle linguistique-sémantique pour identifier automatiquement les données personnelles dans les textes en langage naturel, et de traiter ces données textuelles pour leur sécurité, offuscation et gouvernance. Les outils qui ont été

[7] http://tesniere.univ-fcomte.fr/projet-decript/

développés permettent de répondre aux besoins des entreprises, en ligne avec les exigences de la règlementation RGPD.

Le projet Interreg WebSO+[8] (Plateforme de veille multifonctionnelle WebSO+), en collaboration avec la HEG de Genève, la Haute école de gestion Arc, et les entreprises ERDIL et InnoBridge, propose des méthodes et outils pour assurer la veille et l'intelligence économique par l'analyse linguistique des publications à partir des sites web. Ce projet a également donné lieu à plusieurs publications, parmi lesquelles : [El Abed et al., 2022, Cardey and Greenfield, 2018, Jin et al., 2017].

En complément des travaux cités ci-dessus, cette thèse prend la suite d'une série des travaux menés sur le traitement sémantique et l'exploitation de corpus scientifiques, parmi lesquels :

- La thèse de doctorat de Iana Atanassova [Atanassova, 2012] qui porte sur l'extraction d'informations à partir d'articles scientifiques à des fins de recherche d'information. Elle se base sur l'approche de l'Exploration Contextuelle [Desclés, 2006] qui propose une organisation des marqueurs linguistiques sous forme d'indicateurs qui déclenchent des règles contextuelles.

- Les travaux sur l'analyse des données spatiales dans les articles scientifiques [Atanassova et al., 2015], et plus généralement les travaux sur l'analyse sémantique des articles scientifiques [Atanassova, 2019].

[8] http://tesniere.univ-fcomte.fr/projet-webso/index.html

1.8 Outils d'extraction et d'annotation des expressions temporelles

L'extraction et l'annotation des expressions temporelles permettent de structurer et d'interpréter le contenu temporel des textes. Cette section vise à offrir une perspective approfondie sur les nombreux outils d'extraction des expressions temporelles. Nous passerons en revue plusieurs solutions, en mettant l'accent sur leurs caractéristiques, domaines d'application, ainsi que leurs avantages et inconvénients respectifs.

1.8.1 HeidelTime

HeidelTime est un outil spécialisé pour l'extraction et la normalisation des expressions temporelles dans les textes [Strötgen and Gertz, 2010]. Originellement développé à l'Université de Heidelberg, ce système se distingue par sa capacité à traiter des documents provenant de diverses sources, en rendant possible l'annotation temporelle sur des textes allant des articles d'actualité aux documents historiques ([Strötgen et al., 2013], [Strötgen et al., 2014], [Manfredi et al., 2014], etc).

HeidelTime, en tant qu'outil d'annotation, repose principalement sur des règles pour détecter et normaliser les expressions temporelles. L'outil utilise des transducteurs finis, des patrons lexicaux et une vaste base de données de règles pour identifier les mentions temporelles dans les textes. Une fois ces mentions identifiées, HeidelTime procède à leur normalisation.

Outre ces capacités d'extraction et de normalisation, HeidelTime est également équipé pour gérer les références relatives, s'appuyant sur les métadonnées du texte pour détecter des expressions telles que 'la semaine prochaine' ou 'l'année dernière'.

L'un des atouts majeurs de HeidelTime réside dans sa polyvalence linguistique. L'outil est capable de traiter une variété de langues, bénéficiant d'une internationalisation grâce à la contribution de la communauté scientifique.

Avantages :

- Basé sur des règles, HeidelTime offre une précision considérable lorsqu'il est confronté à des structures linguistiques connues.
- La capacité de traiter une multitude de langues et de types de textes lui confère une polyvalence notable.
- L'outil est en constante évolution grâce à une communauté active, garantissant des mises à jour régulières et des améliorations.

Inconvénients :

- Comme tout système basé sur des règles, HeidelTime peut éprouver des difficultés face à des tournures linguistiques inédites ou exceptionnelles.
- Sa dépendance à une base de règles exige une mise à jour et un entretien constants pour maintenir sa performance à un niveau optimal.

1.8.2 NLTK

Le Natural Language Toolkit (NLTK) est une bibliothèque destinée au traitement du langage naturel conçue pour le langage de programmation Python. Depuis son introduction, NLTK s'est établi comme un outil essentiel pour l'instruction en linguistique computationnelle et en traitement du langage naturel. Au-delà de ses fonctions principales, NLTK est enrichi de divers corpus et ressources lexicales, renforçant sa pertinence pour l'analyse linguistique et la recherche académique [Loper and Bird, 2002].

Bien que NLTK ne soit pas principalement axé sur les expressions temporelles, son architecture modulaire et ses composants diversifiés facilitent cette tâche. Sa capacité à tokeniser, étiqueter grammaticalement et évaluer des structures syntaxiques offre la possibilité de créer des pipelines pour détecter et analyser les expressions temporelles. De plus, plusieurs des ressources lexicales intégrées à NLTK peuvent servir à repérer des termes relatifs au temps.

NLTK trouve son application dans l'enseignement, la recherche et la création de prototypes. Qu'il s'agisse d'analyse de sentiments, de reconnaissance d'entités nommées, de tokenisation ou d'analyse syntaxique, NLTK propose les instruments adaptés à ces enjeux [Bird, 2006].

Avantages :

- Bibliothèque dotée de multiples ressources et outils pour diverses tâches en traitement du langage naturel.
- Particulièrement adaptée à la recherche et à l'éducation en raison de ses corpus intégrés et ressources lexicales.
- Conception modulaire permettant une adaptation aisée à différents besoins.

Inconvénients :

- Face à des outils plus récents tels que SpaCy, NLTK peut montrer ses limites pour des tâches à grande échelle.
- Peut présenter une courbe d'apprentissage pour les novices, vu la diversité de ses outils et méthodes.
- Pour des missions ciblées telles que l'annotation temporelle, des ajustements manuels et une programmation avancée sont nécessaires.

1.8.3 SpaCy

SpaCy se distingue comme une bibliothèque majeure en traitement du langage naturel (NLP) et comme NLTK, SpaCy est une libraire dans Python. Elle est valorisée pour sa rapidité, son efficacité, et sa compétence à analyser d'importantes quantités de texte à des fins industrielles. SpaCy est structurée pour faciliter la transition entre la recherche et le développement [Vasiliev, 2020].

Bien que SpaCy soit équipé pour diverses fonctions en NLP, de la tokenisation à l'analyse de dépendance, sa capacité intrinsèque à gérer les expressions temporelles reste modeste.

Cependant, cette capacité peut être augmentée grâce à des extensions telles que "dateparser"[9]. En utilisant cette synergie, SpaCy est en mesure d'identifier, de normaliser et de décomposer des expressions temporelles variées. Offrant une prise en charge pour un éventail de langues, de l'anglais, au chinois[10], SpaCy est modulable pour s'adapter à différents genres textuels, qu'il s'agisse d'articles, de textes littéraires ou académiques. Avec un modèle adéquat, SpaCy peut également être utilisé pour des langues ou dialectes particuliers.

Avantages :

- Capacité à traiter efficacement de vastes volumes de données [Partalidou et al., 2019].
- Ensemble varié de fonctionnalités NLP prêtes à l'emploi.
- Support pour un large panel de langues avec une extensibilité pour des besoins ciblés.
- Intégration aisée avec d'autres outils et extensions, comme "dateparser".

Inconvénients :

- Nécessité d'extensions ou de modèles additionnels pour des tâches précises.
- Les performances peuvent fluctuer selon les modèles et les données d'apprentissage.
- Les débutants pourraient nécessiter un temps d'adaptation à l'API et aux méthodologies, notamment lors de l'intégration d'extensions.

[9] https://pypi.org/project/dateparser/
[10] https://spacy.io/usage/models

1.8.4 CoreNLP

Stanford CoreNLP est un ensemble d'outils en traitement du langage naturel. CoreNLP est perçu favorablement dans le milieu académique [Song and Chambers, 2014]. Cette suite propose des modules d'analyse linguistique qui coopèrent pour fournir un éventail d'annotations, de la tokenisation à l'analyse de dépendance [Manning et al., 2014]. Un composant clé de CoreNLP axé sur la dimension temporelle est SUTime. Incorporé à Stanford CoreNLP, SUTime est conçu pour la reconnaissance et la normalisation des expressions temporelles dans les textes en anglais [Chang and Manning, 2012]. SUTime est basé sur des règles construites à partir d'expressions régulières.

Avantages :

- Couverture extensive de diverses tâches en NLP.
- Efficacité de SUTime dans la reconnaissance et la normalisation des expressions temporelles.

Inconvénients :

- Mise en place et configuration potentiellement plus exigeantes par rapport à des alternatives comme SpaCy ou NLTK.
- Malgré l'existence de modules pour d'autres langues, ceux-ci peuvent manquer de la solidité ou de l'exhaustivité du module anglais.

1.9 Conclusion de l'état de l'art

Les expressions temporelles jouent un rôle fondamental dans la compréhension des textes, fournissant un cadre de référence qui lie les événements et les informations dans un contexte temporel. Les outils d'extraction et d'annotation de ces expressions sont essentiels pour une multitude de tâches en traitement automatique des langues, allant de la compréhension des textes à l'extraction d'informations au visualisation des données.

Dans ce chapitre, nous avons exploré un éventail d'outils comme HeidelTime, aux outils polyvalentes comme SpaCy, NLTK, et CoreNLP. Chacun de ces outils présente ses propres atouts, qu'il s'agisse de précision, d'adaptabilité ou de support linguistique. Le choix entre eux dépendra des besoins spécifiques du projet en question.

Malgré les avancées significatives dans ce domaine, des défis persistent, notamment en termes de granularité, d'expressions temporelles implicites, et de prise en charge des langues moins courantes. Toutefois, il est impératif de souligner que même ces outils avancés peuvent rencontrer des difficultés à détecter et à normaliser des expressions temporelles complexes. Des expressions telles que 'between January 2, 2019 and March 25, 2020' , 'in the early spring of 1999' ou 'during several days in January 2020' posent des défis. Ces outils ont tendance à identifier deux instances distinctes d'expressions temporelles qui expriment une durée entre deux dates précises plutôt que de reconnaître la durée énoncée dans sa globalité. Dans le cadre de notre projet de recherche, nous avons développé un outil basé sur le schéma d'annotation TimeInfo, capable de capturer ces expressions temporelles complexes et de pallier certaines de ces limitations.

Chapitre 2

Création d'un corpus d'articles scientifiques

Le Traitement Automatique des Langues (TAL) a connu un développement important ces dernières décennies. L'un des aspects cruciaux de la recherche en TAL est la disponibilité de corpus annotés. Les corpus de données textuelles peuvent être utilisés dans de nombreux domaines, tels que la recherche en linguistique, en psychologie, en sociologie, en science politique et bien sûr en traitement automatique du langage naturel. Les corpus de données textuelles nous aident à comprendre la structure de la langue, le contexte d'utilisation des mots, les règles grammaticales, etc. [Hunston, 2022]. Et récemment avec la disponibilité des GPU et TPU, les corpus de données textuelles sont souvent utilisés pour entraîner des modèles d'apprentissage automatique à effectuer diverses tâches en TAL telles que l'étiquetage de parties du discours, la reconnaissance d'entités nommées, l'analyse de sentiments et la traduction automatique [Chowdhary, 2020]. Dans ces cas, la qualité et la taille du corpus annoté déterminent la précision du modèle entraîné. Il est donc essentiel de construire des corpus annotés de qualité.

Mais avant d'obtenir ces corpus annotés, nous devons analyser des données brutes, dans la majorité des cas ces données sont non structurées. L'analyse des données implique l'identification de motifs et de structures dans un corpus de texte donné. Elle peut être effectuée

manuellement ou à l'aide d'outils automatisés. Dans ce chapitre, nous présentons dans un premier le libre accès, la science ouverte et les avantages que présente le libre accès pour le Traitement Automatique des Langues. Ensuite, nous discutons de l'importance de l'analyse de corpus dans la recherche en TAL. Dans un deuxième temps, nous abordons le choix du corpus et l'analyse manuelle des articles scientifiques.

2.1 Le libre accès et le traitement automatique des langues

Le libre accès désigne le principe selon lequel les informations et les ressources sont disponibles de manière gratuite [Suber, 2012]. Cela signifie que tout utilisateur, qu'il soit chercheur, étudiant ou membre du grand public, peut accéder à des articles scientifiques, des livres, des données, des logiciels ou d'autres types de ressources numériques sans avoir à payer des frais d'abonnement ou à obtenir une permission spécifique [Willinsky, 2006]. Le libre accès vise à améliorer la diffusion et l'utilisation des connaissances, à favoriser la collaboration et la réutilisation des travaux, et à renforcer l'équité et l'inclusion dans le domaine de la recherche et de l'éducation [Pöschl and Koop, 2008].

Un aspect du libre accès qui a connu une grande notoriété ces dernières années est l'Open Science ou la science ouverte. La science ouverte vise à rendre la recherche scientifique et ses résultats accessibles à tous, que cela soit les données de recherche, les résultats, les logiciels, ou les articles scientifiques en plein texte [Bertin and Atanassova, 2012]. Les pratiques de la science ouverte visent à accroître la transparence et la collaboration, permettant ainsi aux chercheurs du monde entier de contribuer à la recherche scientifique avec leur expertise et leurs connaissances [Spellman et al., 2018]. En donnant accès libre aux données de la recherche, on favorise non seulement la validation des résultats existants mais on ouvre également la voie à de nouvelles perspectives d'investigation [Munafò, 2016].

Les corpus de données en libre accès peuvent également être utilisés afin d'améliorer la précision des méthodes d'extraction et d'annotation existantes. En effet, avec l'analyse de grandes quantités de textes en langage naturel, les chercheurs peuvent identifier les motifs et les exceptions qui n'auraient peut-être pas été apparents dans des ensembles de données plus

petits. Cela peut conduire au développement de règles plus sophistiquées et nuancées pour mieux gérer la complexité et la variabilité des textes en langage naturel et cela ne serait pas possible sans le libre accès aux données. En plus, la science ouverte en TAL nous donne la possibilité de télécharger et d'utiliser de grands ensembles de données pour l'entraînement et l'évaluation des modèles d'apprentissage automatique. Cela permet aux chercheurs de tester et de développer des modèles plus robustes et plus précis [Sonnenburg et al., 2007].

Nous pouvons conclure que le libre accès a transformé la façon dont la recherche est menée et diffusée. Et par la disponibilité massive des données, le libre accès a eu un impact significatif sur le domaine du Traitement Automatique des Langues.

2.2 Jeu de données CORD-19

Le choix de notre corpus de travail s'est porté sur le COVID-19 Open Research Dataset (CORD-19)[1]. CORD-19 est un ensemble de données qui a été développé par une collaboration d'organisations, dont l'Allen Institute for AI, l'initiative Chan Zuckerberg, l'université de Georgetown, Microsoft Research et la bibliothèque nationale de médecine des National Institutes of Health. Il se compose d'une collection d'articles scientifiques, de prépublications et de communications de conférences liés au SARS-CoV-2 et à d'autres coronavirus. Le jeu de données a été créé en utilisant des techniques de Traitement Automatique des Langues (TAL) pour identifier et agréger des contenus pertinents à partir de diverses sources, notamment PubMed, bioRxiv, medRxiv et la base de données COVID-19 de l'Organisation Mondiale de la Santé [Wang et al., 2020].

Les articles de CORD-19 sont disponibles dans plusieurs formats, notamment PDF, XML et JSON. En plus des articles en texte intégral, l'ensemble de données comprend également des métadonnées d'articles, telles que des informations sur les auteurs, des dates de publication et des noms de revues. Les 59311 documents couvrent un large éventail de sujets liés à la COVID-19, notamment l'épidémiologie, la virologie, l'immunologie, les essais cliniques

[1] https://www.kaggle.com/allen-institute-for-ai/CORD-19-research-challenge

et les interventions de santé publique[2].

2.2.1 Métadonnées

Les documents du corpus sont accompagnés de métadonnées qui fournissent des informations importantes sur leur contenu et leur contexte. Dans cette partie, nous allons faire une analyse plus approfondie des métadonnées du CORD-19. La version que nous utilisons date du 16 avril 2020 et d'autres mises à jour du corpus existent[3]. Les métadonnées du corpus CORD-19 sont organisées en un ensemble de champs qui fournissent des informations sur chaque document. Le champ "cord_uid" est un identifiant unique pour chaque document. Le champ "sha" contient une empreinte numérique unique pour chaque document qui permet de garantir l'intégrité et l'authenticité du document. Le champ "source_x" indique la source du document, qui peut être PMC, Elsevier, medrxiv, WHO, biorxiv et CZI. Le champ "title" contient le titre du document, tandis que le champ "doi" contient le Digital Object Identifier (DOI) du document. Le champ "pmcid" contient l'identifiant PMC du document et le champ "pubmed_id" contient l'identifiant PubMed du document. De plus, le champ "license" indique la licence du document, qui peut varier en fonction de la source. Le champ "abstract" contient le résumé du document, tandis que le champ "publish_time" contient la date de publication du document. Ensuite, le champ "authors" contient une liste des auteurs du document, tandis que le champ "journal" contient le nom du journal dans lequel le document a été publié. Le champ "Microsoft Academic Paper ID" contient l'identifiant du document attribué par Microsoft Academic, tandis que le champ "WHO Covidence" contient l'identifiant attribué par l'OMS. Enfin, les champs "has_pdf_parse", "has_pmc_xml_parse" et "full_text_file" indiquent si le document a été analysé avec des outils de traitement de texte et si le texte intégral est disponible.

[2] https://www.kaggle.com/datasets/allen-institute-for-ai/CORD-19-research-challenge

[3] https://github.com/allenai/cord19

Exemple :

- **cord_uid** : xqhn0vbp
- **sha** : 1e1286db212100993d03cc22374b624f7caee956
- **source_x** : PMC
- **title** : Airborne rhinovirus detection and effect of ultraviolet irradiation on detection by a semi-nested RT-PCR assay
- **doi** : 10.1186/1471-2458-3-5
- **pmcid** : PMC140314
- **pubmed_id** : 12525263
- **license** : no-cc
- **abstract** : BACKGROUND : Rhinovirus, the most common cause of upper respiratory tract infections... METHODS : We aerosolized rhinovirus in a small aerosol chamber... RESULTS : We obtained positive results from filter samples... CONCLUSION : The air sampling and extraction methodology developed in this study should be applicable to...
- **publish_time** : 2003-01-13
- **authors** : Myatt, Theodore A ; Johnston, Sebastian L ; Rudnick, Stephen ; Milton, Donald K
- **journal** : BMC Public Health
- **Microsoft Academic Paper ID** :
- **WHO Covidence** :
- **has_pdf_parse** : True

- **has_pmc_xml_parse** : True
- **full_text_file** : custom_license
- **url** : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC140314/

La figure 2.1 présente le nombre d'articles par source. Dans le cadre de l'ensemble de données CORD-19, "source" fait référence à la source de l'article ou du document, où l'article a été initialement publié ou rendu disponible : serveur de prépublication ou une autre plateforme de publication. Comme nous pouvons le voir, 56,16% des documents proviennent de PubMed Central (PMC) qui est un référentiel numérique gratuit de la littérature de revues biomédicales et des sciences de la vie en texte intégral géré par "US National Library of Medicine (NLM)" et "National Institutes of Health" (NIH)[4]. En seconde position, avec 38,17% des documents proviennent de Elsevier qui est une société d'édition académique spécialisée dans l'édition de revues scientifiques et techniques, de livres et de bases de données. Elsevier publie des articles scientifiques dans des domaines tels que la médecine, les sciences sociales, les sciences de la vie, les sciences physiques et les mathématiques[5].

En plus de la "source", nous avons "journal", un terme qui se réfère au nom de la revue dans laquelle un article a été initialement publié ou mis à disposition. De nombreux articles de l'ensemble de données proviennent de revues scientifiques évaluées par des pairs, mais il y a aussi des articles provenant de serveurs de prépublication, de comptes rendus de conférences et d'autres sources. Dans les métadonnées du CORD-19, nous avons identifié 6220 revues, dont les plus fréquentes sont présentées dans la figure 2.1.

La revue avec le plus de publications est "Journal of Virology". Cette dernière est une revue scientifique dans le domaine de la virologie qui publie des recherches de pointe sur les virus et les maladies virales. La revue couvre un large éventail de sujets, notamment les interactions entre les virus et les hôtes, la pathogenèse virale, le développement de vaccins et les thérapies antivirales. Le "Journal of Virology" est publié par "American Society for

[4] https://www.ncbi.nlm.nih.gov/pmc/
[5] https://www.elsevier.com/

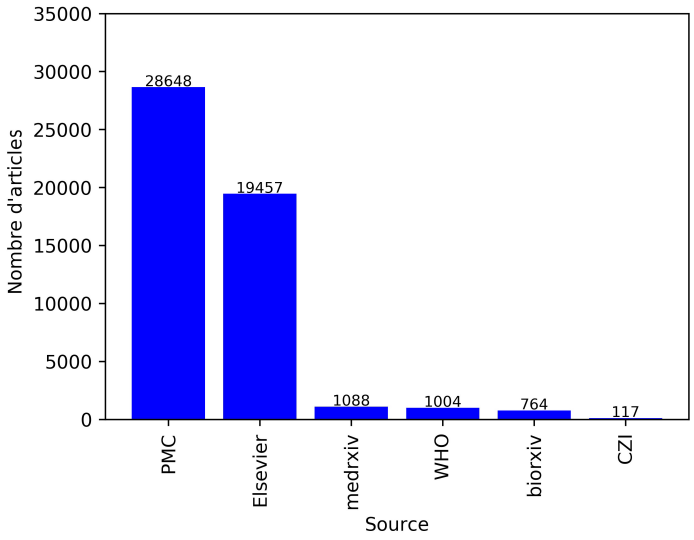


FIGURE 2.1 : Métadonnées CORD-19 : nombre de documents par source 16/04/2020

Microbiology"[6]. En seconde position vient la revue Plos One[7] avec 1569 publications. PLoS One est une revue scientifique multidisciplinaire, qui se distingue par son modèle de publication axé sur le libre accès. PLoS One est publié par la Public Library of Science (PLoS)[8], une organisation à but non lucratif qui vise à rendre les connaissances scientifiques disponibles gratuitement à tous. La portée de la revue est large, couvrant des sujets allant de la recherche fondamentale aux études appliquées, et elle publie des articles scientifiques, des revues et des éditoriaux.

Un autre élément des métadonnées des documents est la date de publication de chaque article. La figure 2.2, présente le nombre d'articles publiés par année dans CORD-19. Même si ce corpus ne contient pas l'intégralité des travaux sur ces pathogènes, nous pouvons distinguer l'émergence des épidémies de SRAS, de MERS et de Covid-19 qui ont conduit à une augmentation significative du nombre de publications dans le corpus qui sont axées sur la compréhension de ces maladies, de leurs origines, de leur transmission et des traitements potentiels. Après l'épidémie de SARS en 2002-2003 [Anderson et al., 2004], il y a

[6] https://asm.org/
[7] https://journals.plos.org/plosone/
[8] https://plos.org/

eu une augmentation du nombre d'articles publiés. De même, l'épidémie de MERS en 2012 [Al Hajjar et al., 2013] a conduit à une augmentation de la recherche scientifique, avec des centaines d'articles publiés sur le virus et les sujets connexes. La pandémie de Covid-19, qui est apparue en 2019, a eu un impact encore plus important, avec des milliers d'articles publiés dans l'année suivant son apparition. L'augmentation rapide des publications scientifiques liées à ces épidémies met en évidence l'importance de la recherche scientifique dans la lutte contre les maladies infectieuses émergentes et dans le développement de stratégies efficaces pour contrôler et prévenir leur propagation [Wang et al., 2020].

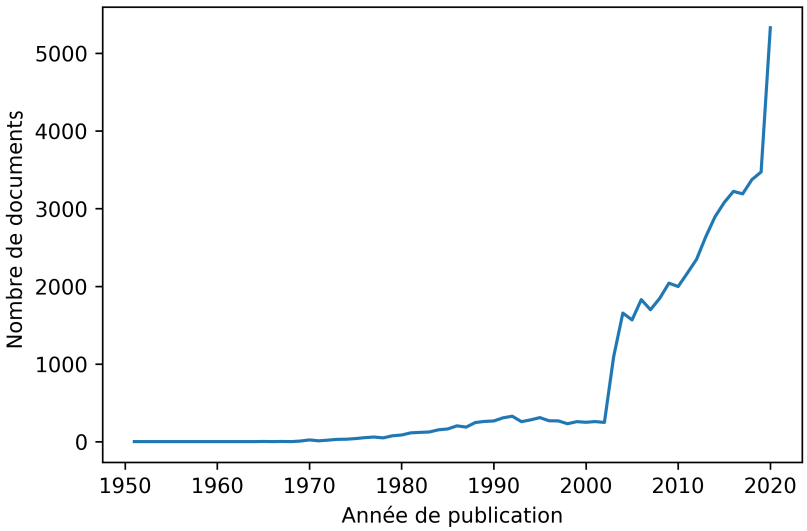


FIGURE 2.2 : CORD-19 Nombre de publications par année

Au sein du corpus CORD-19, à la date du 16 avril 2020, plus de 41 000 articles en texte intégral étaient disponibles. Les articles sont codés au format JSON, ce qui donne accès à la fois au texte intégral et aux métadonnées des articles, comme l'identifiant unique de chaque article, le journal, le ou les auteurs, etc.

- **paper_id** : Identifiant unique pour l'article.
- **metadata** : Métadonnées sur l'article, telles que le titre et les auteurs.
- **body_text** : Le corps principal de l'article, divisé en sections et en paragraphes.

- **ref_entries** : Références à d'autres articles, identifiés par des ID de référence.
- **back_matter** : Toute information supplémentaire à la fin de l'article.
- **bib_entries** : Références aux sources citées dans l'article, identifiées par des ID de citation.

2.2.2 Exemple d'un article

Ici, nous allons voir un exemple d'un article scientifique [Moorthy et al., 2020] au format JSON extrait du corpus CORD-19.


```
{
    "paper_id"  "PMC7047033"
    "metadata"  {
        "title"  "Data sharing for novel coronavirus (COVID-19)"
        "authors"  [
            {
                "first"  "Vasee"
                "middle"  []
                "last"  "Moorthy"
                "suffix"  ""
                "email"  null
                "affiliation"  {}
            } ...]
    }
    "body_text"  [
        {
            "text"  "Rapid data sharing is the basis for public health action. The report
                from the 30 January 2020 International Health Regulations (2005) Emergency
                Committee regarding the outbreak of novel coronavirus (COVID-19) stressed
                the importance of the continued sharing of full data with the World Health
                Organization (WHO). The information disseminated through peer-reviewed
                journals and accompanying online data sets is vital for decision-makers.1\
                u20133 For example, the release of full viral genome sequences through a
                public access platform and the polymerase chain reaction assay protocols
                that were developed as a result made it possible to accurately diagnose
                infections early in the current emergency."
            "cite_spans"  [
                {
                    "start"  436
                    "end"  437
                    "mention"  "1"
                    "ref_id"  "BIBREF0"
                }
                {
                    "start"  438
                    "end"  439
                    "mention"  "3"
                    "ref_id"  "BIBREF2"
```

```
                }
            ]
            "section"  ""
            "ref_spans"  []
        }
        {
            "text"  "Given the many unanswered questions on the reservoir, transmission,
                consequences and manifestations of COVID-19 infection and associated disease
                , our goal is to encourage all researchers to share their data as quickly
                and widely as possible. With this protocol for immediate online posting, we
                are providing another means to achieve immediate global access to relevant
                data. By submitting their studies to \u201cCOVID-19 Open,\u201d researchers
                can share their data while meeting their need to retain authorship, document
                 precedence and facilitate international scientific cooperation in the
                response to this emergency."
            "cite_spans"  []
            "section"  ""
            "ref_spans"  []
        }
    ]
    "ref_entries"  {}
    "back_matter"  []
    "bib_entries"  {
        "BIBREF0"  {
            "title"  "Data sharing in public health emergencies: a call to researchers."
            "authors"  []
            "year"  2016
            "venue"  "Bull World Health Organ"
            "volume"  "94"
            "issn"  "3"
            "pages"  null
            "other_ids"  {
                "DOI"  [
                    "10.2471/BLT.16.170860"
                ]
            }
        } ..
        }
    }
}
```

L'article met en évidence le rôle crucial du partage des données pour combattre efficacement la COVID-19. Il évoque la mise à disposition en Libre Accès des séquences intégrales du génome viral, facilitant ainsi un diagnostic précis dès les premiers instants de la crise. Par ailleurs, les auteurs insistent sur la nécessité d'apporter des réponses aux interrogations persistantes liées au virus et à la maladie qu'il provoque. Ils incitent fortement les chercheurs à diffuser leurs données de manière proactive et exhaustive.

En ce qui concerne la structure, les métadonnées de l'article ci-dessus est présenté sous la forme d'un document JSON. Le fichier est structuré sous la forme d'un objet comprenant plusieurs clés associées à leurs valeurs respectives. L'élément 'paper_id' est une chaîne de caractères qui est un identifiant unique. De plus, la clé 'metadata' contient des informations sur l'article, notamment son titre et ses auteurs, qui sont représentés sous la forme d'un objet de type liste. Chaque auteur est représenté par un objet comprenant des informations telles que le prénom, le nom et l'affiliation. Le corps du texte de l'article est représenté sous forme d'une liste d'objets. Chaque objet correspond à un paragraphe et contient une clé 'text' qui représente le contenu du paragraphe en question. Les citations dans le texte sont également représentées sous forme d'objets, avec des clés pour l'indice de début et de fin de la citation, ainsi que la référence bibliographique associée. La section des références bibliographiques est représentée par un objet 'bib_entries' qui contient des clés pour chaque référence bibliographique, avec les informations correspondantes telles que le titre de l'article, les auteurs, le lieu de publication et les identifiants tels que le DOI.

Le Traitement Automatique des Langues (TAL) souligne l'importance des corpus annotés et le mouvement vers le libre accès et la science ouverte renforce cette dynamique, facilitant la diffusion et l'accessibilité des connaissances. C'est dans ce contexte que la valeur d'un corpus tel que CORD-19 se manifeste, démontrant la synergie possible entre le libre accès, les besoins spécifiques de la recherche et les avancées en TAL.

Chapitre 3

Méthodologie de catégorisation Syntaxico-Sémantique de l'information temporelle TimeInfo

Nous appelons notre méthode Syntaxico-Sémantique car nous ajoutons une valeur sémantique à l'information temporelle et cela grâce à l'attribut 'interval' que nous allons voir plus en détails ci-après. Et syntaxique car nous essayons d'indiquer l'élément syntaxique qui introduit l'information temporelle, ces éléments syntaxiques sont équivalents, mais pas identiques aux éléments déclencheurs que nous trouvons dans [Ferro et al., 2003].

Dans ce chapitre, nous présentons notre schéma d'annotation des expressions temporelles que nous avons nommé TimeInfo. Ce schéma est conçu dans le but de fournir une représentation sémantique des expressions temporelles extraites des articles scientifiques. Lors de l'élaboration de TimeInfo, l'une des priorités a été d'assurer l'interopérabilité avec les schémas et formats d'annotation existants. Nous avons opté pour une syntaxe à la fois accessible et flexible, en utilisant le langage de balisage extensible XML (la figure 3.2 représente le document DTD[1] qui illustre les spécifications structurales nécessaires à la modélisation et à la validation des documents XML du schéma TimeInfo). De plus, nous avons privilégié l'utili-

[1]Document type definition : https ://doi.org/10.3917/ela.137.0073

sation d'un maximum d'éléments et d'attributs issus de TIMEX2 et TIMEX3 lorsque cela a été possible. De ce fait, TimeInfo peut être utilisé par un annotateur humain, pour développer des outils de détection d'extraction et d'annotation sémantique des données temporelles et aussi, pour la visualisation des données et l'ajout d'une dimension temporelle aux moteurs de recherche. Nous présentons d'abord les étapes qui nous ont permis de développer le schéma TimeInfo, notamment, l'analyse des données du corpus CORD-19 et l'extraction manuelle des phrases contenant une expression temporelle pertinente. Ensuite, nous présentons une méthode à base de règles pour l'extraction, la catégorisation et l'annotation sémantique de l'information temporelle au sein des articles scientifiques en se basant sur le schéma d'annotation TimeInfo.

3.1 Analyse manuelle du corpus

L'analyse manuelle d'un corpus de textes scientifiques est une méthode courante pour extraire des informations et identifier des tendances ou des éléments pertinents. L'objectif de cette analyse était d'extraire des expressions temporelles. Cette analyse des données commence par la lecture linéaire des articles afin de comprendre comment les expressions temporelles sont présentées dans les textes scientifiques. Nous avons analysé manuellement notre corpus d'articles scientifiques dans le but d'extraire des expressions temporelles. Cette analyse des données commence par la lecture linéaire des articles afin de comprendre comment les expressions temporelles sont présentées dans les textes scientifiques. Puis, nous avons segmenté le corpus en phrases et nous les avons considérées comme unités textuelles de base pour la suite de notre analyse. Les phrases sont des unités qui sont relativement indépendantes et qui peuvent être analysées du point de vue de leur syntaxe avec les outils existants en TAL. Ainsi, les éléments clés tels que les sujets, les verbes et les objets peuvent être identifiés, ce qui peut s'avérer utile pour des tâches telles que l'Étiquetage de Rôles Sémantiques [Màrquez et al., 2008] et l'Analyse des Dépendances [Chen and Manning, 2014]. De plus, l'analyse de phrases peut être automatisée et des outils de segmentation automatique tels que

SpaCy[2] et NLTK[3], deux bibliothèques Python avec un module de segmentation de texte en phrases qui sont facilement accessibles et permettent de traiter de grandes masses de données textuelles.

Cependant, le travail uniquement sur les phrases peut présenter des inconvénients, notamment la perte des relations entre les phrases elles-mêmes, ce qui peut limiter la compréhension globale d'un texte. De plus, la qualité des données peut limiter l'analyse de phrases, car les données textuelles peuvent contenir des erreurs de syntaxe, des coquilles ou d'autres incohérences, ce qui peut rendre l'analyse de phrases plus difficile et impacter le résultat des outils de segmentation de phrases.

Durant cette partie de notre recherche, aucun outil d'extraction, de segmentation ou d'annotation des données n'a été utilisé. Nous avons procédé à une analyse visuelle des données pour nous familiariser avec le format et le contenu des articles de notre corpus. Pour constituer notre corpus d'entrée (voir tableau 3.1), le choix des articles scientifiques était arbitraire et la lecture était linéaire, l'extraction de la phrase se faisant seulement si une information temporelle est détectée. À la fin de cette analyse, nous avons choisi 101 phrases, issues de 58 articles scientifiques traitant de SARS-CoV et SARS-CoV-2 et contenant des informations temporelles pertinentes. Nous tenons à souligner que les 58 articles ne sont pas les seuls que nous avons analysés. En effet, plusieurs articles ne contiennent aucune information temporelle pertinente.

Puis, nous avons construit un premier corpus de phrases annotées manuellement, à l'instar du tableau 3.1 avec quatre colonnes : l'identifiant de l'article scientifique, l'identifiant de la phrase, le contenu de la phrase et l'expression temporelle. L'identifiant de l'article scientifique est utilisé pour l'identification de la source et le référencement et l'identifiant de la phrase est un numéro unique attribué à chaque phrase pour faciliter son identification.

Cette organisation nous permet en premier lieu d'extraire et de visualiser les expressions temporelles séparément de leurs cotextes[4]. Aussi, nous pouvons revenir facilement à la phrase

[2] https://spacy.io/

[3] https://www.nltk.org/

[4] Cotexte ou co-texte en linguistique, environnement linguistique immédiat d'un texte. source : https ://www.universalis.fr/dictionnaire/cotexte/ 18/11/2023

source pour voir l'interaction de l'expression temporelle avec son cotexte.

3.2 Conception du schéma TimeInfo

En termes de structure, TimeInfo reprend la plupart des informations présentes dans les systèmes précédents (TIMEX2 et TIMEX3) et introduit de nouveaux attributs qui permettent des distinctions plus fines entre les expressions temporelles et une représentation plus riche.

La Figure 3.1 présente un diagramme de TimeInfo, montrant tous les éléments et toutes les valeurs possibles qu'un attribut peut prendre. En rouge, nous avons représenté les attributs présents dans les éléments de TIMEX2 ou TIMEX3. En bleu, nous avons représenté les nouveaux attributs spécifiques à notre cadre d'annotation TimeInfo.

Contrairement à TIMEX2 et TIMEX3, TimeInfo permet de représenter des expressions temporelles complexes, telles que "from December 2001 to April 2002", en les reconnaissant comme des intervalles de temps avec leurs divers attributs (granularité, durée, précision, etc.). Une telle expression serait analysée par TIMEX3 en tant qu'une étendue de texte contenant deux dates diférentes qui sont "December 2001" et "April 2002", alors qu'il s'agit en réalité d'un intervalle temporelle.

Le noyau de TimeInfo repose sur la valeur sémantique de l'information temporelle et de son cotexte. Bien que TimeInfo soit initialement conçu pour la langue anglaise, il peut être adapté à d'autres langues, vu que la valeur sémantique d'une expression temporelle ne change pas d'une langue à l'autre.

Dans notre système d'annotation de l'information temporelle représenté dans la figure 3.1, nous pouvons observer les différents attributs et leurs valeurs. Dans la section suivante, nous analysons les attributs qui compose la balise TIMEINFO, aussi, nous donnons des exemples de phrases annotées avec notre système d'annotation de l'information temporelle.

TABLE 3.1 : Exemples de phrases avec expressions temporelles

idArticle	idPhrase	Phrase	TempData
PMC261870	2	This pan-viral microarray was used as part of the global effort to identify a novel virus associated with severe acute respiratory syndrome (SARS) in March 2003, as reported by Ksiazek et al. (2003).	in March 2003
PMC517714	6	A total of 54 SARS-CoV genomic sequences (37 from the public database prior to October 14, 2003 and 17 sequenced within our institute) are used in our current analysis.	prior to October 14, 2003
PMC517714	7	Taking the SARS-CoV isolated from palm civet cat as the putative originating SARS-CoV, our calculations suggest that the earliest possible date for SARS emergence is predicted to be Oct 21, 2002.	is predicted to be Oct 21, 2002
PMC138858	12	Vibrio cholerae O1 has figured prominently in the history of infectious diseases as a cause of periodic global epidemics, an affliction of refugees in areas of social strife and as the disease first subjected to modern epidemiological analysis during the classic investigations of John Snow in mid-19th century London.	mid-19th
PMC520756	11	Records were available from December 2001 to April 2002 ; 13 such investigations during this period resulted in identification of 62 household contacts, all of which were contacted ; out of 38 workplace/social contacts identified, 32 were contacted (84%).	from December 2001 to April 2002
PMC544195	22	As of August 12, 2003 there were 438 probable and suspected cases of Severe Acute Respiratory Syndrome (SARS) in Canada – the majority located in Toronto.	As of August 12,2003
PMC512294	25	Numerous malaria epidemics have occurred in western Kenya, with increasing frequency over the past 20 years.	over the past 20 years
PMC1456961	36	The need for improved access to high quality public health (PH) information has been echoed in various forums involving public health professionals, librarians, and information professionals since the mid 1990s.	since the mid 1990s

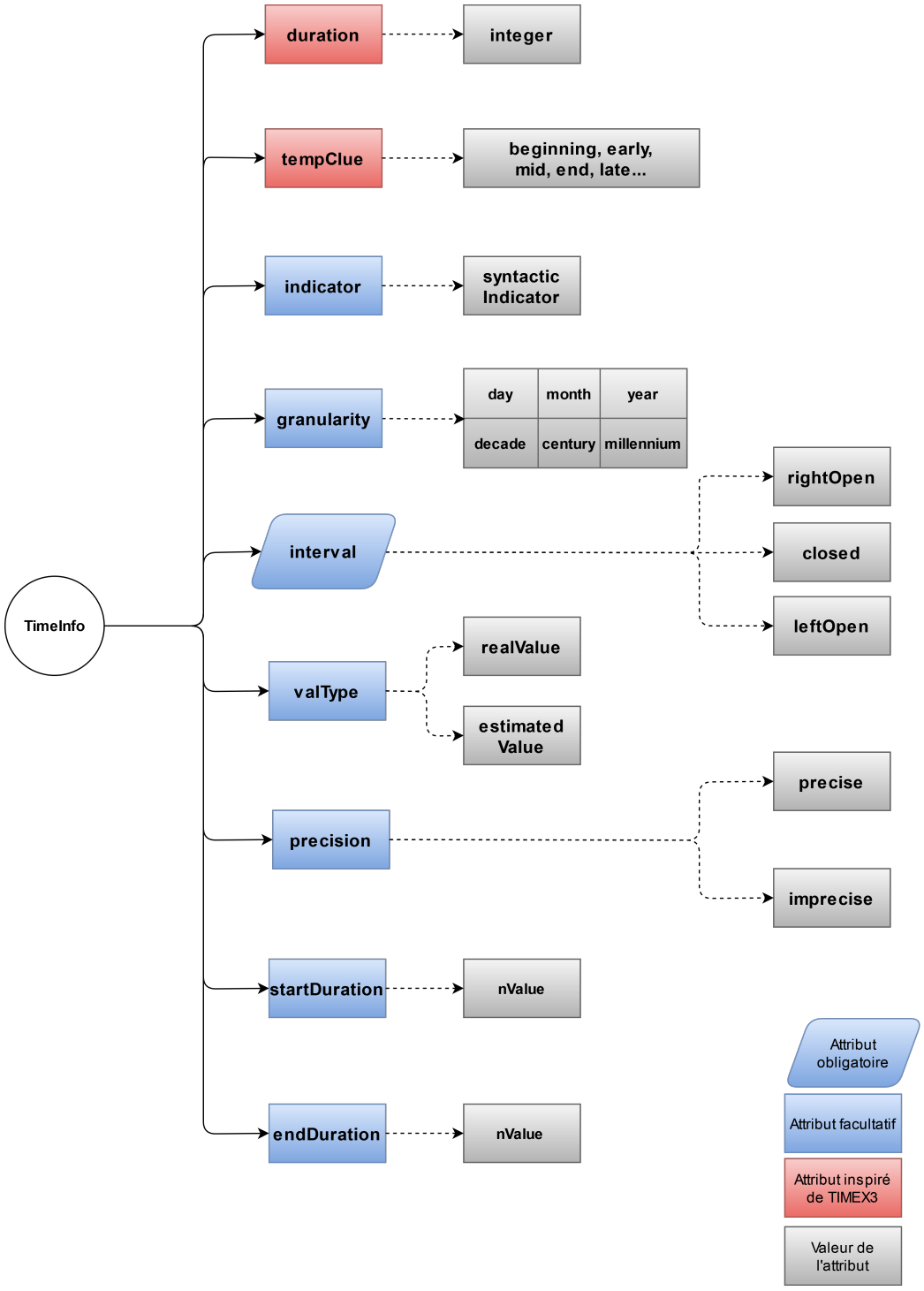


FIGURE 3.1 : Schéma d'annotation TimeInfo

```
<!ELEMENT TimeInfo (#PCDATA)>
<!ATTLIST TimeInfo
    interval (closed | leftOpen | rightOpen | closed_duration) #
       REQUIRED
    granularity (day | month | year | decade | century |
       millennium) #IMPLIED
    nValue CDATA #IMPLIED
    duration CDATA #IMPLIED
    startDuration CDATA #IMPLIED
    endDuration CDATA #IMPLIED
    indicator CDATA #IMPLIED
    precision (precise | imprecise) #IMPLIED
    tempClue CDATA #IMPLIED
    valType (realValue | estimatedValue) #IMPLIED
>
```

FIGURE 3.2 : Document type definition (DTD) du schéma TimeInfo

3.2.1 L'attribut interval

L'attribut `interval` au sein du schéma TimeInfo est obligatoire. Cette exigence découle de son rôle de vecteur sémantique permettant d'indiquer le type d'intervalle temporel représenté par les données en question. Sa présence facilite ainsi la compréhension des données temporelles et leur interprétation. `Interval` possède trois valeurs possibles : `closed`, `left-open` et `right-open`[5]. Un intervalle est considéré comme fermé si l'expression nous permet d'identifier à la fois le début et la fin de l'intervalle temporel. De plus, un intervalle avec la valeur fermé peut décrire une date ou une durée, comme le montrent les exemples suivants : 'On January 2020, SARS-CoV-2 was isolated and announced as a new, seventh, type of human coronavirus' [Bzówka et al., 2020]. 'The surveys were conducted from February 1, 2020 to February 10, 2020, as transmission of COVID-19 peaked across China and stringent interventions were in place.' [Zhang et al., 2020]

Pour illustrer les intervalles `left-open` et `right-open`, nous pouvons considérer

[5]En mathématiques, il existe quatre types d'intervalles : fermé, ouvert à gauche, ouvert à droite et ouvert. Pour notre schéma d'annotation, nous avons considéré uniquement trois de ces types d'intervalles, en laissant de côté les intervalles ouverts. En effet, nous n'avons pas pu observer d'occurrences d'intervalles ouverts dans nos ensembles de données de publications scientifiques.

une représentation du temps comme une ligne droite d'un point A à un point B (voir la figure 3.3). Dans un intervalle `right-open`, la valeur du point A est exprimée et la valeur du point B n'est pas exprimée. Par exemple : 'In Asia, several media outlets have opted to use "Wuhan-pneumonia" 7 instead of COVID-19 in their reporting even though WHO has explicitly advised against naming new human infectious diseases with geographic locations or populations since 2015.' [Lin, 2020]. Dans un intervalle `left-open`, le point de départ A est inconnu ou non identifiable à partir de l'expression linguistique, et le point d'arrivée B est connu. Par exemple : 'Based on epidemiological data before 2019, only six CoVs proved to cause human respiratory diseases : i) HKU1, HCoV-NL63, HCoV-OC43 and HCoV-229E only lead to mild upper respiratory disease, but rarely bring about severe diseases in people ; ii) SARS-CoV and MERS-CoV attack lower respiratory tract and always induce se-vere respiratory syndrome.' [Kang et al., 2020]

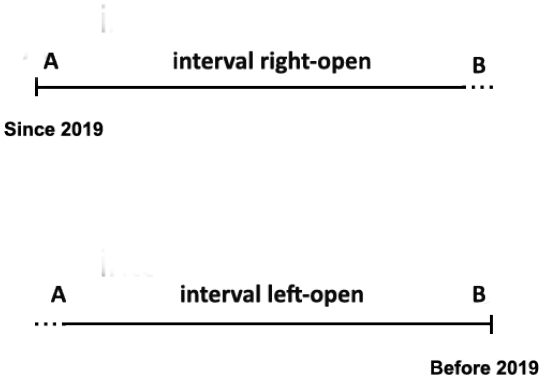


FIGURE 3.3 : Intervalle right-open et left-open

3.2.2 L'attribut granularity

Le deuxième attribut que nous introduisons est la granularité. La granularité représente la plus petite unité temporelle exprimée dans une phrase donnée. Elle prend comme valeur : `"day"`, `"month"`, `"year"`, `"decade"`, `"century"` ou `"millennium"`. Par exemple, dans l'expression '12 January 1992', la valeur de la granularité est `day`. Par défaut, la plus petite unité que nous utilisons est `"day"`, et ce choix a été motivé par les occurrences présentes dans notre corpus qui traite de SARS-CoV et SARS-CoV-2. Des unités plus petites peuvent être considérées pour d'autres ensembles de données.

3.2.3 nValue

L'attribut `nValue` est équivalent à `value` de la balise `TIMEX3` du schéma d'annotation ISO-TimeML [Pustejovsky et al., 2010]. Tout comme `value`, `nValue` sert à normaliser les variétés linguistiques des expressions temporelles en une forme canonique, permettant ainsi une interprétation uniforme et lisible par la machine. À titre d'exemple, l'expression 'le 10 octobre 1954' serait représentée par la valeur `1954-10-10`. Cette méthode standardisée simplifie la comparaison et le traitement automatisé des données temporelles.

3.2.4 Les attributs duration, startDuration et endDuration

L'attribut `duration` donne le nombre de jours, de mois ou d'années que couvre l'intervalle temporel. L'information sur la durée d'un événement peut être utile dans le contexte de la recherche d'informations ou à des fins de visualisation de données.

Les attributs `startDuration` et `endDuration` fournissent le début et la fin d'un intervalle temporel. La valeur de `startDuration` et `endDuration` est standardisée pour faciliter le traitement avec l'outil informatique sous la forme `YYYY-MM-DD` (voir les exemples annotés dans la figure 3.4).

3.2.5 L'attribut indicator

L'attribut `indicator` recueille les éléments qui introduisent les données temporelles. Ces éléments sont extraits du cotexte. Les valeurs de l'attribut `indicator` ne sont pas limitées à des ensembles fermés, mais peuvent être n'importe quelle expression linguistique, ou une liste d'expressions qui introduisent les données temporelles dans le texte, telles que les prépositions, par exemple "de ... à", ou les phrases adverbiales comme "vers la fin de". La présence de l'attribut `indicator` vise à faciliter le processus de construction d'algorithmes pour l'annotation des expressions temporelles. Les éléments syntaxiques donnés par l'attribut `indicator` peuvent être utilisés soit comme des caractéristiques pour les algorithmes d'apprentissage automatique, soit pour développer des ressources linguistiques et des règles pour la détection et l'annotation des expressions temporelles. Par ailleurs l'attribut `indicator` est destiné à l'annotateur humain ou à toute personne qui souhaite améliorer les systèmes de détection de l'information temporelle en analysant la syntaxe d'un texte donnée. Un tel attribut n'est pas nécessaire pour l'utilisateur final d'un programme informatique, de ce fait l'utilisation de cet attribut n'est pas obligatoire dans notre schéma d'annotation de l'information temporelle.

Reprenons l'exemple : 'A total of 54 SARS-CoV genomic sequences (37 from the public database **prior to October 14, 2003** and 17 sequenced within our institute) are used in our current analysis.' [Vega et al., 2004]. Ici nous avons l'adjectif `prior` suivie de la préposition `to` qui introduisent l'information temporelle `October 14, 2003`. Nous catégorisons cette dernière dans l'intervalle `left-open`. Pour des exemples annotés, voir la figure 3.4.

3.2.6 Les attributs precision et tempClue

Certaines expressions linguistiques indiquent des données temporelles pour lesquelles les limites (début et fin d'un événement dans le temps) ne peuvent pas être précisément identifiées. Par exemple, l'expression 'in the mid-19th' indique un intervalle 'flou' où aucune année spécifique ne peut être considérée comme un début ou une fin de l'intervalle.

La valeur de `precision` peut être `precise` comme dans l'expression 'October 14, 2003' ou 'imprecise' comme dans 'early December'. Lorsque l'expression temporelle est `imprecise`, elle est souvent introduite par un adjectif tel que 'early', 'mid', 'late', etc. L'attribut `tempClue` stocke de tels adjectifs qui indiquent des parties de l'intervalle. Par exemple, pour l'expression 'in the mid-19th century', la valeur de `tempClue` est `mid`. Les attributs `precision` et de `tempClue` peuvent tous deux servir de fonctionnalités dans un moteur de recherche sophistiqué où des données temporelles précises sont recherchées. Par exemple, la requête : 'Covid-19 cases before the end of July 2021' devrait, théoriquement, récupérer des données avec des mentions telles que 'early 2021' ou 'in the early 2020s'.

3.2.7 L'attribut valType

En analysant notre corpus, nous avons identifié deux types d'expressions temporelles différents : les valeurs estimées et les valeurs réelles. Par exemple, dans 'Taking the SARS-CoV isolated from palm civet cat as the putative originating SARS-CoV, our calculations suggest that the earliest possible date for SARS emergence **is predicted to be Oct 21, 2002**.', l'information temporelle est une valeur estimée, tandis que l'expression 'the date of emergence was January 12, 1992' indique une valeur temporelle réelle. L'attribut `valType` exprime cette distinction. Il est facultatif et peut prendre deux valeurs : `estimated value` et `real value`. Le but ici est de séparer les valeurs réelles des valeurs estimées et cette information est pertinente du point de vue de la recherche d'informations.

```
Our data includes 1212 patients ranging<TimeInfo interval="closed"
   granularity="day" duration="25" startDuration="2020-01-21" endDuration
   ="2020-02-14" indicator="from-to" precision="precise" valType="
   realValue">from January 21 to February 14, 2020</TimeInfo>, and
   covering 18 regions of Henan province.\cite{wang2020epidemiological}
```

```
Taking the SARS-CoV isolated from palm civet cat as the putative
   originating SARS-CoV, our calculations suggest that the earliest
   possible date for SARS emergence <TimeInfo interval="closed"
   granularity="day" nValue="2002-10-21" indicator="is predicted to be"
   valType="estimatedValue"> is predicted to be Oct 21, 2002</TimeInfo>.'
    \cite{vega2004mutational}.
```

```
<TimeInfo interval="right-open" granularity="month" startDuration="
   2019-12-XX"
 "indicator="Since the end of"  valType="realValue" precision="imprecise"
     tempClue="end">Since the end of December 2019</TimeInfo> the Chinese
     city of Wuhan has reported a novel pneumonia caused by coronavirus
    disease 2019 (COVID-19), which is spreading domestically and
    internationally.' \cite{lai2020factors}.
```

```
The Severe Acute Respiratory Syndrome (SARS) was first reported <TimeInfo
    interval="closed" granularity="month" nValue="2002-11-XX" indicator="
   in" valType="realValue" precision="precise">in November 2002</TimeInfo
   > and rapidly spread to a number of distant global regions <TimeInfo
   interval="left-open" granularity="year"  endDuration="2003-XX-XX"
   indicator="by"  valType="realValue" precision="imprecise" tempClue="
   early"> by early 2003</TimeInfo>.' \cite{vega2004mutational}.
```

FIGURE 3.4 : Exemples d'expressions temporelles annotées avec TimeInfo

3.3 Remarques conclusives sur TimeInfo

TimeInfo permet de capturer à la fois la valeur sémantique des informations temporelles grâce à des attributs tels que `interval`, ainsi que les éléments syntaxiques qui introduisent ces informations, dans un format interopérable.

L'analyse manuelle de notre corpus a révélé l'importance de comprendre la présentation des expressions temporelles dans les articles scientifiques. Cette approche a permis de constituer un corpus de phrases annotées, vérifié manuellement, mettant en lumière les interactions entre les expressions temporelles et leurs contextes dans les textes scientifiques.

TimeInfo propose une représentation des expressions temporelles plus riche que les autres schémas d'annotation, en particulier pour les intervalles complexes. Son applicabilité ne se limite pas à l'anglais, mais peut s'étendre à d'autres langues, compte tenu de l'universalité de la valeur sémantique des expressions temporelles.

Chapitre 4

Développement de règles linguistiques d'annotation pour TimeInfo

4.1 Construction des règles pour la catégorisation et l'annotation de l'information temporelle avec TimeInfo

La détection d'informations temporelles dans les textes s'appuie dans de nombreux travaux sur la reconnaissance d'entités nommées (REN). La REN est une sous-tâche de l'extraction d'informations qui cherche à localiser et classer les entités nommées dans le texte en catégories prédéfinies telles que les noms de personnes, les organisations, les lieux et les expressions temporelles. Il existe plusieurs outils basés sur les entités nommées pour la détection des expressions temporelles, notamment, spaCy[1], NLTK[2], CoreNLP[3], etc. Bien que ces outils aient montré leur utilité dans la détection d'informations temporelles, ils ont également leurs limites. Par exemple, ils peuvent avoir des difficultés à détecter et à interpréter les expressions temporelles complexes telles que 'from February 21, 2020 to March 18, 2020' et 'since the end of January to February 2020'. De tels outils se concentrent principalement sur la détec-

[1] https://spacy.io/
[2] https://www.nltk.org/
[3] https://stanfordnlp.github.io/CoreNLP/

tion d'entités et ne prennent pas toujours en compte le contexte sémantique plus large dans lequel les expressions temporelles sont utilisées.

Dans ce contexte, la valeur ajoutée de notre approche basée sur TimeInfo réside dans le fait que, plutôt que de se concentrer uniquement sur la REN, nous proposons d'appliquer un ensemble de règles linguistiques pour identifier et interpréter les expressions temporelles dans les textes. Ainsi, dans cette section, nous présentons la conception et l'application d'un ensemble de motifs, ou `patterns`, pour identifier des dates spécifiques, des durées, ainsi que des intervalles fermés et ouverts. Nous explorons également comment l'utilisation des règles linguistiques peut aider à améliorer la précision de la reconnaissance des expressions temporelles.

Notre approche s'appuie sur des règles pour identifier les expressions temporelles en fonction de leur structure syntaxique. Ces règles ont été élaborées en tenant compte des diverses façons dont une expression temporelle peut être formulée, et elles visent à capturer les expressions temporelles dans toute leur complexité et diversité. Afin d'illustrer de manière plus concise et visuelle le processus décrit ci-dessous, voir la figure 4.1.

L'analyse des expressions temporelles a constitué notre point de départ. Nous avons entrepris une étude approfondie en examinant des centaines d'exemples. La phase suivante a été axée sur l'identification des prépositions et des phrases adverbiales. Plus précisément, nous avons ciblé les locutions prépositionnelles associées à chaque type d'intervalle. Les locutions identifiés sont listées pour être utilisées dans les ensembles de motifs. Parallèlement, une attention particulière a été accordée aux exceptions et aux locutions prépositionnelles qui pourraient présenter une ambiguïté, par exemple : 'The data has been considered for Indian region **from 30-Jan-2020 onwards** [...].' Ici l'intervalle de l'expression temporelle est `right_open`, alors que dans 'We included consecutive patients with diagnosis of COVID-19 pneumonia or heart failure [...] **from Dec. 1 to Feb. 28, 2020**' l'intervalle est `closed_duration`. Autres exemples : '**since March 14 to March 20, 2020**' et '**since January 25, 2020**' où la première expression temporelle est catégorisée avec l'intervalle `closed_duration` la deuxième avec l'intervalle `right_open`.

Une règle linguistique est composée de plusieurs éléments dont la fonction syntaxique, la

position du token, la forme canonique des mots (lemmes), la présence ou l'absence d'expressions dans le contexte gauche ou droit. Pour une description approfondie de la méthode mise en œuvre, voir la section 4.2. Nous avons créé plusieurs ensembles de motifs pour détecter et catégoriser l'information temporelle en fonction des différents intervalles de TimeInfo. Le cotexte de l'information temporelle a été le pilier sur lequel nous avons basé cette catégorisation.

L'implémentation des règles s'est concrétisée par le développement d'un programme dédié. Parallèlement, un corpus annoté a été généré, consolidant ainsi notre approche.

4.2 Implémentation des règles linguistiques

L'implémentation des motifs et des règles a été faite avec spaCy qui permet d'écrire nos propres règles, en utilisant le module 'Matcher', ainsi que le module 'EntityRuler'. Le processus commence par la définition des motifs dans un fichier JSON. Chaque motif est une liste de dictionnaires, où chaque dictionnaire représente un token et ses attributs. Par exemple,

```
1 {"LOWER"  "january"}
```

correspond à un token dont le texte en minuscules est 'january'. Les motifs peuvent également inclure des opérations, comme

```
1 {"IS_DIGIT"  True  "OP"  "?"}
```

qui correspond à un token optionnel qui est un chiffre.

Une fois les motifs définis, nous les chargeons dans notre programme et les utilisons pour créer un Matcher spaCy pour chaque type d'expression temporelle. Pour chaque Matcher, nous ajoutons les motifs correspondants et lui assignons une étiquette qui correspond à la catégorie de l'expression temporelle. Ensuite, nous utilisons le module EntityRuler de spaCy pour ajouter les entités reconnues par les Matchers à la pipeline de traitement du langage naturel. Pour chaque document, le Matcher parcourt chaque token et vérifie si une séquence de tokens correspond à l'un des motifs. Si c'est le cas, il crée une entité pour cette séquence et lui attribue l'étiquette correspondante.

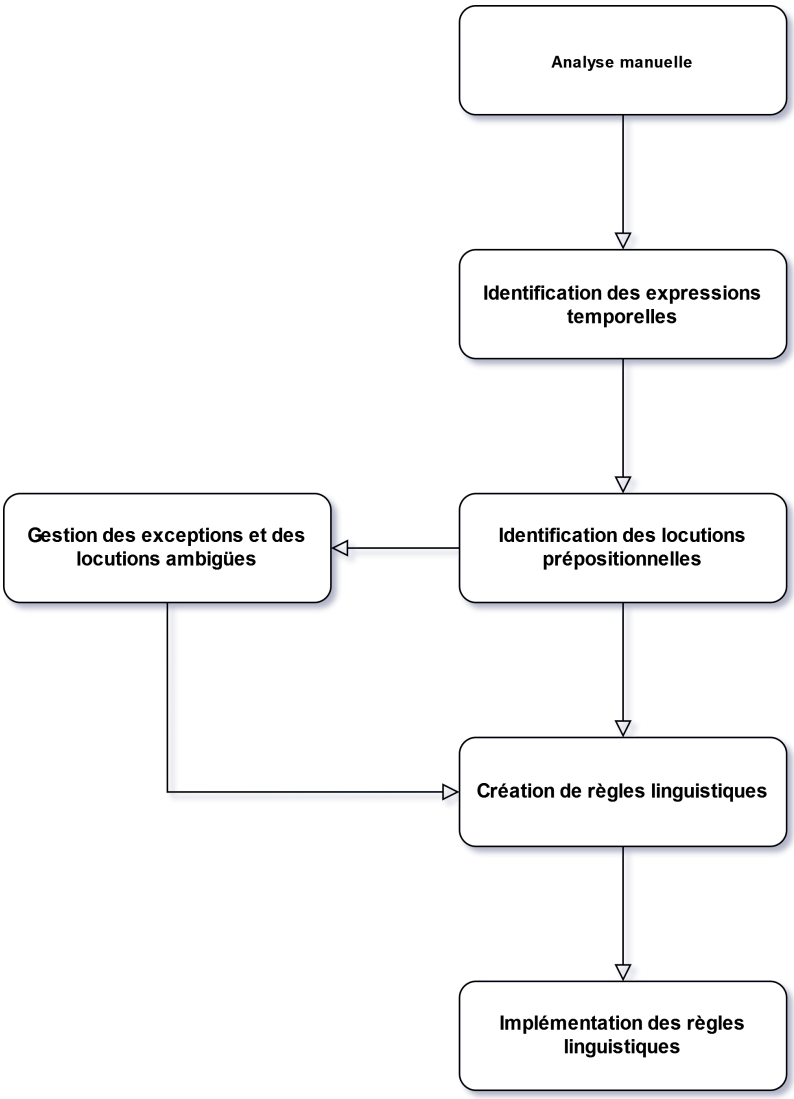


FIGURE 4.1 : Les étapes de création des règles d'annotation pour TimeInfo

Par exemple, étant donné le texte "The contract was signed on the 21st of March 2022", un Matcher avec le motif de date identifierait "the 21st of March 2022" comme une date et lui attribuerait l'étiquette "DMY".

En utilisant le Matcher et l'EntityRuler de spaCy de cette manière, nous pouvons créer un système robuste pour la détection des expressions temporelles, capable de gérer une grande variété de constructions linguistiques.

4.2.1 Détection des expressions temporelles

Dans cette partie, nous présentons notre détecteur de dates, qui fonctionne de la même manière que les outils de reconnaissance d'entités nommées classiques. Il commence par détecter une expression temporelle et lui attribut l'étiquette 'DATE' ou 'TIME'. Pour notre outil, nous avons choisi d'utiliser l'étiquette 'DMY' pour 'Day', 'Month' et 'YEAR'. Cependant les jours, mois et années ne sont pas les seules expressions temporelles que notre outil détecte. En effet, il permet la détection de plusieurs variations d'expressions temporelle, notamment les décennies et les siècles. Pour la détection des intervalles nous ajoutons des règles syntaxique que nous allons voir dans les sous-sections ci-après.

Dans l'exemple 4.2 le motif correspond à un token dont le texte correspond à l'expression régulière fournie. Cette expression régulière est conçue pour correspondre aux noms complets ou abrégés des jours de la semaine en anglais, comme "Monday" ou "Mon".

```
1 {'label'  'DMY'
2  {'TEXT'  {'REGEX'  '\b(? Mon(? day)?|Tue(? sday)?|Wed(? nesday)?|Thu(? rsday)?|Fri(? day)?|
       Sat(? urday)?|Sun(? day)?)\b'}}]}
```

FIGURE 4.2 : Ensemble de motif utilisé pour la détection des jours de la semaine

L'exemple 4.3 est un ensemble de motifs composé de trois parties :

- `{'POS': 'ADJ', 'OP': '?'}` : Cela correspond à un token optionnel (OP : '?' signifie que le token peut apparaître zéro ou une fois) qui est un adjectif (POS : 'ADJ'). Cela pourrait permettre de reconnaître des expressions comme 'early 20th century'

(début du 20ème siècle) où ‘early’ (début) est un adjectif qui modifie ‘20th century’ (20ème siècle).

- `'TEXT':{'REGEX':'\b\d{1,2}(?:st|nd|rd|th)?\s?(?:[cC]entury)\b'}` : Le motif est expression régulière conçue pour identifier les siècles , comme ‘20th century’ ou ‘21st Century’. La partie d1,2(? :st|nd|rd|th)? correspond à un chiffre de un à deux chiffres, éventuellement suivi par ‘st’, ‘nd’, ‘rd’, ou ‘th’, et la partie `\s?(?:century|Century)` correspond à un espace optionnel suivi par ‘century’ ou ‘Century’.

- `{'TEXT': {'REGEX':'\b\d{1,2}(?:st|nd|rd|th)?-?[cC]entury\b'}}` : Cette expression régulière est similaire à la précédente, mais elle est conçue pour correspondre aux siècles écrits avec un trait d’union, comme ‘20th-century’.

```
{
  "label"  "DMY"
  "pattern"  [
    {
      "POS"  "ADJ"
      "OP"  "?"
    }
    {
      "TEXT"  {
        "REGEX"  "\b\d{1,2}(?:st|nd|rd|th)?\s?(?:[Cc]entury)\b"
      }
    }
    {
      "TEXT"  {
        "REGEX"  "\b\d{1,2}(?:st|nd|rd|th)?[-][Cc]entury\b"
      }
    }
  ]
}
```

FIGURE 4.3 : Ensemble de motifs utilisé pour la détection des siècles

L’exemple 4.4 représente un motif plus complexe pour la détection des expressions temporelles telles que ‘3rd of January 2022’ ou ‘early January 2022’. Cet ensemble de motifs est composé de plusieurs éléments :

- `{'POS': 'ADJ', 'OP': '?'}` : Cela correspond à un token optionnel qui est un adjectif tels que 'mid', 'early', 'end', etc.

- `{'TEXT': \{'REGEX': '\b\d\{1,2\}(?:st|nd|rd|th)?\b'\}
  , 'OP': '?'}`
  : expression régulière utilisée afin de d'identifier à un jour du mois, comme : '1st', '2nd', '3rd', '4th', jusqu'à '31st'.

- `{'IS\SPACE': true, 'OP': '?'}`:
  ce motif correspond à un espace optionnel. Il faut noter que nous n'avons pas besoin de spécifier les espaces entre les éléments, l'outil de spaCy s'en charge automatiquement. Cependant, lors de l'analyse manuelle du corpus, nous avons remarqué que deux espaces peuvent être présents dans certains cas.

- `{'TEXT': \{'REGEX':'\b(?:Jan(?:uary)?|Feb(?:ruary)?`
  `|Mar(?:ch)?|Apr(?:il)?|May|Jun(?:e)?`
  `|Jul(?:y)?|Aug(?:ust)?|Sep(?:tember)?`
  `Oct(?:ober)?|Nov(?:ember)?|Dec(?:ember)?)\b'\}\}`
  Cette expression régulière est conçue pour identifier les mois de l'année.

- `{'TEXT': {'REGEX': '\textbackslash b(?:19|20)\\d{2}\ b'}`
  `,'OP': '?'}` Cette expression régulière est conçue pour identifier une année du 20ème ou du 21ème siècle, comme "1990" ou "2022. Cette expression peut-être modifiée selon les besoins et les corpus.

Ainsi, nous avons construit plusieurs ensembles de motifs ou de 'patterns' afin d'identifier le plus d'expressions temporelles possibles. Nous utilisons cet outils avec d'autres éléments linguistique afin de détecter les expressions temporelles des différents intervalles, '`closed`', '`closed_duration`', '`left_open`' et '`right_open`'.

```
{
  "label"  "DMY"
  "pattern"  [
    {
      "POS"  "ADJ"
      "OP"  "?"
    }
    {
      "TEXT"  {
        "REGEX"  "\b\d{1,2}(?:st|nd|rd|th)?\b"
      }
      "OP"  "?"
    }
    {
      "IS_SPACE"  true
      "OP"  "?"
    }
    {
      "TEXT"  {
        "IN"  [
          "of"
        ]
      }
      "OP"  "?"
    }
    {
      "IS_SPACE"  true
      "OP"  "?"
    }
    {
      "TEXT"  {
        "REGEX"  "\b(?:Jan(?:uary)?|Feb(?:ruary)?|Mar(?:ch)?|Apr(?:il)?|May|Jun(?:e)?|
            Jul(?:y)?|Aug(?:ust)?|Sep(?:tember)?|Oct(?:ober)?|Nov(?:ember)?|Dec(?:ember)
            ?)\b"
      }
    }
    {
      "IS_SPACE"  true
      "OP"  "?"
    }
    {
      "TEXT"  {
        "REGEX"  "\b\d{1,2}(?:st|nd|rd|th)?\b"
      }
      "OP"  "?"
    }
    {
      "IS_SPACE"  true
      "OP"  "?"
    }
    {
      "TEXT"  {
        "REGEX"  "\b(?:19|20)\\d{2}\b"
      }
      "OP"  "?"
    }
  ]
}
```


FIGURE 4.4 : Ensemble de motifs pour la détection d'une date

4.2.2 Détection de l'intervalle closed et closed_duration

Dans cette partie, nous présentons des exemples de motifs pour l'identification des intervalle fermés. Nous avons deux sous catégories dans les intervalles fermés, '`closed`' et '`closed_duration`'.

L'intervalle fermé '`closed`' est le plus simple a détecter vu qu'il comporte une seule unité temporelle, par exemple 'in 1992', 'In December 2019', 'in early 20th-century', etc. Afin de correctement détecter ces intervalles, nous utilisons notre outil de reconnaissance d'entités nommées, en y ajoutant d'autres ensembles de motifs, tels que les prépositions qui introduisent les intervalles fermés. La principale difficulté pour la détection des expressions temporelles avec un intervalle '`closed`' est que plusieurs exceptions doivent être gérées afin de ne pas introduire du bruit. Nous pouvons observer ces exceptions dans la figure 4.5.

```
[
    {'LEMMA'  {'IN'  ['on'  'in'  'be']}}
    {'ENT_TYPE'  'DMY'  'OP'  '+'}
    {'TEXT'  {'NOT_IN'  ['-'  'to'  'and']}}
    {'TEXT'  {'NOT_IN'  {'ENT_TYPE'  'DMY'}}}
]
```

FIGURE 4.5 : Exemple d'un ensemble de motifs avec des exceptions

Dans cet exemple, nous retrouvons **'DMY'** suivi de l'opérateur **'+'** , ce qui nous permet de détecter une date donnée. Ce motif doit obligatoirement être précédé par les prépositions **'in'** ou **'on'** ou par le verbe **'to be'**. Ici, nous utilisons la fonction **'LEMMA'** pour mettre **'to be'** à sa forme canonique, ce qui n'est pas nécessaire pour les prépositions qui sont invariables. Ensuite, le motif **{'TEXT' : {'NOT_IN' : ['-', 'to', 'and']}}** spécifie que le token ou l'élément qui vient après **'DMY'** ne doit pas être l'une des chaînes spécifiées '**-**', '**to**' et '**and**'. Enfin, le motif **{'TEXT' : {'NOT_IN' : {'ENT_TYPE' : 'DMY'}}}** évite la détection d'une seconde occurrence de l'entité **'DMY'**. Ainsi, dans ces ensembles de motifs, nous ne détectons pas les expressions telles que 'December 1 st , 2013 to April 30 th , 2014', 'between 11 February 2020 and 13 February 2020', '2020-2021', etc.

Avec ces différentes règles nous pouvons détecter des expressions temporelles qui ont un

intervalle fermés, par exemples :

- 'According to this fit, the date of a substantial reduction in the number of cumulative positive cases in Italy (below 100 cases) **is April 22, 2020**'.
- 'This has been exemplified by the novel coronavirus, known as SARS-CoV-2, which was first identified as the cause of an outbreak of pneumonia in Wuhan, China, **in December 2019**, and rapidly spread around the world 1-3 .'
- 'The maximum likelihood (ML) tree, inferred from the full genome viral sequences available **on March 3 rd , 2020** (Supplementary Figure S1), showed a well-supported cluster of European and Asian sequences (reported in Figure 2a), which contained a subclade (subclade A, Figure 2a) including a sequence isolated in Germany that .'

La seconde catégorie des intervalles fermés, désignée par 'closed_duration', mérite une attention particulière. En effet, les outils de détection des expressions temporelles, développés en se basant sur le schéma d'annotation TIMEX3, n'identifient pas les durées en tant que telles en une seule expression temporelle.

La conception de ces règles a nécessité une analyse approfondie des occurences dans notre corpus. Notre objectif était de comprendre comment les durées, caractérisées par un début et une fin bien définis, sont habituellement présentées dans la littérature académique. À la suite de cette analyse, nous avons pu identifier plusieurs manières possibles d'écrire ces durées.

Dans l'exemple 4.6, nous analysons un ensemble de motifs pour l'identification des durées. Les motifs sont illustrés par des exemples d'expressions temporelles que nous avons identifiées et extraites de notre corpus.

- **{'LOWER' : {'IN' : ['from', 'between']}}** : Cette ligne cherche les mots "from" ou "between", la fonction **LOWER** permet de mettre la chaîne en minuscule.
- **{'ENT_TYPE' : 'DMY', 'OP' : '+'}** : Cette ligne est le deuxième élément qui identifie la première unité temporelle qui est le début d'une durée donnée.

```
[
    {'LEMMA'  {'IN'  ['from'  'between']}}
    {'ENT_TYPE'  'DMY'  'OP'  '+'}
    {'TEXT'  {'IN'  [' '  '-'  '.'  '('  ')']}  'OP'  '?'}
    {'IS_SPACE'  True  'OP'  '?'}
    {'LEMMA'  {'IN'  ['to'  'up'  'and'  'till']}}
    {'IS_SPACE'  True  'OP'  '?'}
    {'ENT_TYPE'  'DMY'  'OP'  '+'}
    {'TEXT'  {'IN'  [' '  '-'  '.'  '('  ')']}  'OP'  '?'}
    {'IS_SPACE'  True  'OP'  '?'}
    {'ENT_TYPE'  'DMY'  'OP'  '?'}
]
```

FIGURE 4.6 : Exemple d'un ensemble de motifs pour détecter un intervalle closed_duration

- **{'TEXT' : {'IN' : [',', '-', '.', '(', ')']}, 'OP' : '?'}** : Cette ligne cherche un token qui est un signe de ponctuation spécifique (virgule, tiret, point, parenthèse ouverte, parenthèse fermée). L'opérateur ? indique que ce token est facultatif et peut apparaître zéro ou une fois.
- **{'IS_SPACE' : True, 'OP' : '?'}** : Cette ligne cherche un espace.
- **{'LOWER' : {'IN' : ['to', 'up', 'and','through', 'till']}}** : Cette ligne cherche à identifier 'to', 'up', 'and', 'through' et 'till'. Des prépositions utilisées afin de capturer la transition vers la fin de l'intervalle de temps, comme dans "from January to March".
- **{'IS_SPACE' : True, 'OP' : '?'}** : Cette ligne cherche à nouveau un espace, qui est facultatif.
- **{'ENT_TYPE' : 'DMY', 'OP' : '+'}** : Cette ligne cherche à identifier la deuxième unité temporelle qui donne la fin d'une durée donnée.

Les règles, que nous utilisons sont suffisamment génériques pour détecter plusieurs variations d'expressions temporelles telles que :

- 'Subjects were pre-screened for eligibility and were randomized **between August 2nd, 2010 and August 13th, 2010**.'

- ‘All data were collected as part of routine clinical practice in 15 outpatient physical therapy clinics of Intermountain Healthcare, located in the Salt Lake City, Utah region **from October 1, 2004 through April 30, 2010**.’
- ‘In the Faroe Islands, off the coast of Denmark, no cases of MS were reported **from 1929 to 1943**.’

Ces règles peuvent également identifier une expression temporelle spécifique, telles que les informations temporelles qui sont introduites par l’expression "during the period of" (voir la figure 4.7).

```
[
      {"LOWER"  "during"}
      { "LOWER"  "the"}
      { "LOWER"  "period"}
      {"LOWER"  "of" }
      {"ENT_TYPE"  "DMY"  "OP"  "+" }
      { "IS_SPACE"  true  "OP"  "?"}
      {'LOWER'  {'IN'  ["through" "to"]}}
      { "IS_SPACE"  true "OP"  "?"}
      { "ENT_TYPE"  "DMY"  "OP"  "+"}
]
```

FIGURE 4.7 : Exemple d’un ensemble de motifs pour identifier une expression temporelle spécifique

Exemples :

- ‘This decision was based on the timing of the initial detection of PEDV in the US (April 2013) and the availability of data summarizing temperature and % RH in shipping containers traveling from Asia to the US **during the period of December 31, 2012 to January 16, 2013** [12] .’
- ‘We collected clinical data for 165 patients with clinically confirmed severe SARS who were hospitalized for treatment in Beijing **during the period of 5 March through 31 May 2003** [6] .’

Nous avons mis en avant quelques patterns de type ‘`closed`’ et ‘`closed_duration`’ utilisés dans la détection des expressions temporelles dans les textes scientifiques.

4.2.3 Détection des intervalles left-open et right-open

Dans cette partie, nous présentons des ensembles de règles et de motifs qui sont utilisés pour la détection, l'extraction et l'annotation des intervalles ouverts. Le schéma d'annotation TimeInfo introduit deux types d'intervalles ouverts : '`left-open`' et '`right-open`'.

L'intervalle '`left-open`' représente des expressions temporelles où l'intervalle de temps est ouvert à gauche, c'est-à-dire le début de la durée n'est pas précisée dans l'information temporelle.

Ci-dessous, nous avons l'exemple 4.8 utilisé pour identifier l'intervalle '`left-open`'.

```
[
  {"LOWER"  {"IN"  ["prior"  "till"  "by"  "until"  "as"  "up"  "before"  "as late as"]}}
  {"LOWER"  {"IN"  ["to"  "of"]  "OP"  "?"}}
  {"LOWER"  "the"  "OP"  "?"}
  {"POS"  "ADJ"  "OP"  "?"}
  {"ENT_TYPE"  "DMY"  "OP"  "+"}
]
```

FIGURE 4.8 : Exemple d'un ensemble de motifs pour identifier l'intervalle left_open

Le pattern 4.8 identifie des expressions qui commencent par l'un des mots spécifiés dans la première ligne (par exemple "prior", "till", "by", etc.), éventuellement suivis par "to" ou "of", puis éventuellement par "the", et enfin par une date. L'opérateur "OP" avec la valeur "?" signifie que l'élément est facultatif, tandis que "+" indique que l'élément peut apparaître une ou plusieurs fois. Exemple d'expression temporelle identifiée : "Australian IBV strains detected **prior to the 1980's**, including Australian vaccine strains, were termed 'classical' IBV strains."

Le 'pattern' 4.9 est conçu pour capturer des expressions temporelles qui commencent par 'as of', suivie d'une date donnée. En combinant les éléments de ce 'pattern' nous pouvons identifier des expressions temporelles telles que 'Though the virus that causes COVID-19 seems to be highly contagious, **as of February 24**, the epidemic had been largely contained within Hubei province.'

L'intervalle '`right-open`' représente des expressions temporelles où l'intervalle de

```
[
  {"LOWER"  "as"}
  {"LOWER"  "of"}
  {"ENT_TYPE"  "DMY"  "OP"  "+"}
]
```

FIGURE 4.9 : Exemple d'un pattern spécifique pour identifier un intervalle left_open

temps est ouvert à droite, c'est-à-dire, la fin de la durée n'est pas précisée dans l'information temporelle. Dans l'exemple 4.10, le 'pattern' est structuré pour rechercher un mot qui est l'une des prépositions listées, notamment "since", "after", etc. Ces prépositions sont couramment utilisées pour indiquer le début d'un intervalle de temps ouvert à droite. Le "pattern" cherche ensuite le determinent "the", qui est facultatif, suivi par un adjectif, qui est également facultatif. Enfin, l'ensemble de motifs cherche une expressions temporelle en utilisant l'entité "DMY".

```
[
  {"LOWER"  {"IN"  ["since"  "after"  "starting"  "following"  "commencing"]}}
  {"LOWER"  "the"  "OP"  "?"}
  {"POS"  "ADJ"  "OP"  "?"}
  {"ENT_TYPE"  "DMY"  "OP"  "+"}
]
```

FIGURE 4.10 : Exemple d'un ensemble de motifs pour identifier l'intervalle right_open

Ainsi, en combinant ces règles, ce "pattern" peut identifier des expressions temporelles telles que :

- 'Southern China has been considered a hypothetical influenza epicenter **since the early 1980's** as several pandemic influenza viruses emerged from this region and there was frequent influenza activity at the human-animal interface (Shortridge and Stuart-Harris, 1982) .'
- '**Since 2011** several outbreaks of equine coronavirus have been identified in adult horses, causing it to be named as an emerging pathogen.'
- 'We excluded South Sudan from these analyses, as FSI data were only available **from 2012 onwards**.'

Chapitre 5

Construction d'un corpus avec expressions temporelles annotées : TimeTank

Dans ce chapitre, nous présentons la conception du corpus TimeTank [Yahiaoui and Atanassova, 2023], spécialement dédié à l'étude des expressions temporelles dans la littérature scientifique.

La création de TimeTank répond à un besoin croissant de disposer d'outils fiables pour l'analyse des expressions temporelles. Nous explorons les différentes étapes de sa conception, nous mettons en lumière les défis rencontrés, les choix méthodologiques adoptés, ainsi que les solutions mises en œuvre pour assurer la pertinence et la qualité du corpus.

5.1 Corpus de données temporelles annotées : TimeTank

TimeTank est un corpus dédié à l'étude des expressions temporelles dans les textes scientifiques. Ce corpus, que nous avons constitué à partir du CORD-19 [Wang et al., 2020], est par l'extraction et l'annotation automatique suivies d'une vérification manuelle. Pour sa dénomination, nous nous sommes inspirés de TimeBank [Pustejovsky et al., 2003b]. qui est un

corpus annoté de textes d'actualité en anglais développé comme partie du projet TimeML. Alors que TimeTank se concentre sur l'annotation des expressions temporelles dans les textes scientifiques, offrant ainsi un contexte différent et une application plus spécialisée. Aussi, TimeTank capture les nuances et spécificités des expressions temporelles ce qui offre une compréhension plus fine de l'information temporelle. Notre corpus est disponible en libre accès sur la plateforme DATA UBFC[1] et Zenodo[2].

Le corpus comprend 1200 expressions temporelles annotées selon le schéma d'annotation TimeInfo [Yahiaoui and Atanassova, 2022]. Les phrases sont tirées de 603 articles scientifiques. Le nombre total de phrases est de 1186, chaque phrase peut contenir plus d'une expression temporelle. Bien que notre notre algorithme a extrait plus 150 000 phrases avec des expressions temporelles annotées, nous avons choisi de travailler sur une sélection de 1200 phrases par un souci de qualité des données. En effet, bien que l'extraction des phrases ait été réalisée automatiquement à l'aide de notre schéma d'annotation TimeInfo, chaque phrase de cet échantillon a été révisée manuellement. Cette étape de révision manuelle assure la précision des annotations temporelles, garantit que chaque phrase est représentative du type d'intervalle temporel auquel elle est attribuée.

Les expressions temporelles dans notre corpus sont classées en quatre types d'intervalles : "right_open", "left_open", "closed" et "closed_duration". Chaque type est représenté par exactement 300 phrases, ce qui assure une représentation équilibrée de diverses formes d'expressions temporelles. Chaque phrase dans le corpus est accompagnée de métadonnées détaillées, y compris l'identifiant de l'article, le titre de l'article, l'identifiant unique de la phrase, l'expression temporelle et le type d'intervalle. Ces métadonnées enrichissent notre corpus en fournissant des informations contextuelles qui permettent de retrouver la source et le contexte initiale de chaque phrase si une analyse plus approfondie des données est nécessaire.

Ci-dessous, des tableaux avec des exemples de phrases du corpus TimeTank qui illustre sa structure et son contenu.

[1] https://search-data.ubfc.fr/
[2] https://zenodo.org/record/8364409

TABLE 5.1 : Exemples de phrases annotées avec l'intervalle "closed"

phrase	temporal_expression
"On December 31, 2019, a total of 27 cases were reported; meanwhile, a rapid response team led by the Chinese Centre for Disease Control and Prevention (China CDC) was formed to conduct detailed epidemiologic and aetiologic investigations in Wuhan."	"On December 31, 2019"
"The first case of the novel coronavirus in South Korea occurred in late January 2020, approximately two months after the first case globally occurred in the Hubei province of China."	"in late January 2020"
"These time lags produced similar results on the case confirmation rate on Feb. 10, 2020 (Table 2)."	"on Feb. 10, 2020"
"The AD mats sampled on April 10th, 2016, looked different from the original observation 175 (less than 3 weeks earlier)."	"on April 10 th , 2016"
"On 23/01/2020 all trains, flights and public transports connecting Wuhan with the outside were suspended."	"On 23/01/2020"

Le tableau 5.1 présente 5 phrases extraites du corpus TimeTank contenant des expressions temporelles annotées avec l'intervalle "closed". Le tableau 5.2 montre 6 phrases annotées avec l'intervalle "closed_duration", indiquant des périodes délimitées par deux bornes temporelles précises. On y trouve des exemples tels que "from Jan. 20 to Mar. 4, 2020" ou "between March 18th 2020 and March 20th 2020". Le tableau 5.3 présente 7 phrases du corpus TimeTank annotées avec l'intervalle "left_open", correspondant à des intervalles ouverts à gauche et fermés à droite. On peut y observer des expressions comme "By 06/03/2020" ou "up to February 10, 2020" qui situent des événements avant une date limite. Le tableau 5.4 contient 6 phrases annotées avec l'intervalle "right_open", ouvert à droite et fermé à gauche. Les expressions comme "since Feb. 12, 2020" ou "after January 23rd 2020" positionnent les événements après une date de départ.

Ces exemples illustrent la diversité des expressions temporelles annotées dans le corpus TimeTank selon la typologie des intervalles temporels définie dans TimeInfo. Dans ces tableaux, nous avons choisi de présenter uniquement 2 colonnes : la phrase et l'expression temporelle, par souci de concision. Cependant, il est important de rappeler que dans la version complète du corpus TimeTank, chaque phrase est accompagnée de métadonnées comprenant

TABLE 5.2 : Exemples de phrases annotées avec l'intervalle closed_duration

phrase	**temporal_expression**
"263 Specifically, for January 22, 2020 to March 15, 2020, we collect daily mean 2-meter temperature (in degrees centigrade), total 2-meter precipitation (in mm), and mean 1000 hPA specific humidity (in kg/kg)."	"for January 22, 2020 to March 15, 2020"
"We used the existing reported data from January 23 to March 20 2020 for observing, performing parameter estimation, and forecasting COVID-19 dynamics in different countries/regions."	"from January 23 to March 20 2020"
"We do this via a nationally representative survey of 3,452 Italian adults between March 18th 2020 and March 20th 2020."	"between March 18th 2020 and March 20th 2020"
"were extracted from the daily briefings on novel coronavirus cases from Jan. 20 to Mar. 4, 2020, provided on the official website of the National Health Commission of the People's Republic of China [5] ."	"from Jan. 20 to Mar. 4, 2020"
"Between August and December 2008, a total of 369 study participants presenting with ILI at the two study centres were recruited."	"Between August and December 2008"
"The dataset includes all confirmed cases in China reported from 31/12/2019 to 23/02/2020."	"from 31/12/2019 to 23/02/2020"

TABLE 5.3 : Exemples de phrases annotées avec l'intervalle left_open

phrase	temporal_expression
"As of 18 March, 2020, 690 cases were announced as infected in the community, including asymptomatic cases, but excluding those abroad in countries such as China or those infected on a large cruise ship : the Diamond Princess [1] ."	"As of 18 March, 2020"
"Shortly after the vaccine was approved, the vaccine adverse-event monitoring system noted by mid-1999 an excess of cases of intussusception among recently vaccinated infants, eventually prompting the vaccine to be withdrawn."	"by mid-1999"
"By 06/03/2020[3], approximately 45 days post introduction, the model suggests that approximately 60% (R 0 = 2.25) and 64% (R 0 = 2.75) of the population would have already been exposed to SARS-CoV-2."	"By 06/03/2020"
"For parameters 1 , 2 , and a, we used the daily cumulative confirmed cases up to February 10, 2020 to retrieve their optimal values to reflect the current dynamics in each city."	"up to February 10, 2020"
"As of 3rd March 2020, 90,870 cases and 3,112 deaths of the disease COVID-19 caused by a novel coronavirus had been reported worldwide 1 ."	"As of 3rd March 2020"
"While it is an open question around the generalisability of the Chinese approach to other jurisdictions [9] , there is also evidence of containment success (as of late March 2020) outside mainland China, from Singapore, Hong Kong and Taiwan [10]."	"as of late March 2020"
"In the travel restriction scenario we assume long term enforcement of individual mobility restrictions (travel was restricted until the end of June 2020)."	"until the end of June 2020"

TABLE 5.4 : Exemples de phrases annotées avec l'intervalle right_open

phrase	**temporal_expression**
"Since 2001, concerns regarding complications of smallpox vaccination and smallpox infection in persons with immunocompromised conditions, such as HIV/AIDS or transplantations [94] , have been discussed by Bartlett and others [95]"	"Since 2001"
"Data analysis reveals that strengthening the public health interventions, tracing imported cases and improving the confirmation rate are effective and timely after January 23 rd 2020."	"after January 23 rd 2020"
"The disruption initiated by COVID-19 was modeled as an ‘intervention‘ starting February 3, 2020."	"starting February 3, 2020"
"A clinically diagnosed case was defined as a suspected case with imaging features of pneumonia, which has only been only applicable in Hubei Province since Feb. 12, 2020 [9]."	"since Feb. 12, 2020"
"Since 8 December, 2019, clusters of pneumonia cases of unknown etiology have emerged in Wuhan City, Hubei Province, China [1, 2] ."	"Since 8 December, 2019"
"Since the late 2000’s, there has been a global increase of respiratory disease outbreaks associated with EV-D68."	"Since the late 2000’s"

des colonnes supplémentaires : l'identifiant unique de l'article scientifique, le titre de l'article, et l'identifiant unique de la phrase dans le corpus. Ces métadonnées permettent de replacer chaque phrase dans son contexte, de retrouver facilement l'article source. Les tableaux 5.1, 5.2, 5.3 et 5.4 illustrent la complexité inhérente aux variantes d'expressions temporelles, une complexité qui a posé un défi significatif lors de la conception de TimeInfo Tagger, notre outil d'extraction et d'annotation de l'information temporelle. Face à cette hétérogénéité, nous avons intégré un ensemble exhaustif de règles linguistiques afin de détecter et de catégoriser les expressions temporelles.

5.2 Processus d'annotation des expression temporelles

L'élaboration d'annotations d'expressions temporelles pour le corpus TimeTank s'est déroulée suivant une démarche en quatre étapes (illustrée dans la figure 5.1).

1. Conceptualisation du schéma d'annotation TimeInfo

   Au préalable à l'annotation, nous avons défini précisément les directives et la structure du schéma TimeInfo [Yahiaoui and Atanassova, 2022]. Ce schéma catégorise sémantiquement les expressions temporelles selon 4 types d'intervalles (left_open, right_open, closed et closed_duration). Cette base théorique était essentielle pour guider et développer les règles d'annotation des expressions temporelles.

2. Développement de règles syntaxiques

   Nous avons ensuite conçu des règles syntaxiques pour détecter les expressions correspondant aux dates et aux intervalles temporels. Ces règles ont permis d'identifier automatiquement un large éventail d'expressions conformes aux intervalles de TimeInfo. Leur précision a été affinée grâce à plusieurs cycles d'essais et de corrections sur des exemples concrets.

3. Annotation automatisée avec le script TimeInfo Tagger

   Les règles syntaxiques ont ensuite été implémentées dans un script Python pour annoter automatiquement les phrases extraites de CORD-19. Ce script explore le corpus, extrait

les phrases pertinentes et génère les annotations d'intervalles temporels. Au total, plus de 150 000 phrases ont été annotées par cette approche automatisée.

4. Vérification manuelle sur un échantillon qualitatif

   Enfin, même si l'extraction et l'annotation initiales étaient automatisées, nous avons effectué une vérification manuelle sur un échantillon qualitatif de 1200 phrases. Cette étape a permis de corriger des erreurs résiduelles et de garantir la fiabilité des annotations temporelles dans le corpus final TimeTank.

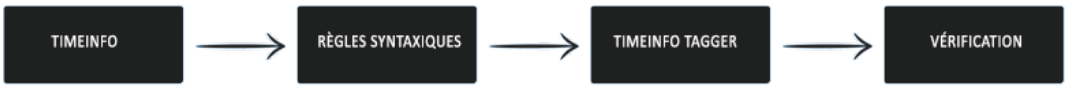


FIGURE 5.1 : Processus d'annotation des expression temporelles du corpus TimeTank

En combinant ces quatre étapes, nous avons pu construire le corpus TimeTank avec une annotation précise des expressions temporelles. Ce processus garantit non seulement la qualité des annotations, mais fournit également une base solide pour toute recherche ou application future utilisant TimeTank.

5.3 Utilisations et perspectives d'enrichissement de TimeTank

5.3.1 Utilisations de TimeTank

Le corpus TimeTank offre des possibilités variées d'utilisation pour la recherche en traitement automatique des langues et en extraction d'informations. Ce corpus fournit un ensemble de phrases annotées avec des expressions temporelles catégorisées selon le schéma TimeInfo. La richesse de ces expressions temporelles structurées ouvre la voie à de multiples applications, que ce soit pour faire progresser la recherche sur la temporalité dans les textes ou pour le développement d'outils exploitant la dimension temporelle dans les textes.

Tout d'abord, le corpus TimeTank peut être exploité conjointement avec des annotations d'événements telles que définies dans la schéma TimeML. En effet, TimeML propose un schéma pour identifier et caractériser les événements dans les textes, en plus de l'annotation des expressions temporelles. Coupler l'annotation des événements TimeML avec les expressions temporelles de TimeTank sur un même corpus permettrait une analyse fine des événements. Cet enrichissement mutuel offrirait une compréhension approfondie de la structure temporelle des connaissances véhiculées dans les articles scientifiques. L'interopérabilité entre TimeInfo et TimeML ouvre donc des perspectives prometteuses pour des études avancées des relations entre temporalités et événements.

Aussi, les chercheurs peuvent exploiter TimeTank pour développer et évaluer des approches d'extraction automatique d'expressions temporelles. Par exemple, un échantillon des phrases annotées peut servir à entraîner un modèle d'apprentissage machine supervisé afin de détecter les entités temporelles dans des textes scientifiques. De plus, TimeTank peut constituer le fondement d'analyses linguistiques sur l'usage contextuel des expressions temporelles dans les publications.

Par ailleurs, ce corpus autorise la conduite d'analyses quantitatives par le biais de méthodes de fouille de textes. L'extraction automatique de statistiques sur les distributions tem-

porelles peut révéler des tendances et des évolutions dans les références temporelles au fil du développement des connaissances sur la COVID-19. Ces mesures peuvent être visualisées sous forme de timelines interactives offrant une vue d'ensemble sur la structuration temporelle du discours scientifique.

L'intégration des métadonnées temporelles de TimeTank dans des moteurs de recherche d'information spécialisés améliorerait leur pertinence pour des requêtes comportant des critères chronologiques. Un système de questions-réponses sur la COVID-19 pourrait ainsi exploiter ces repères temporels pour mieux répondre à des demandes du type "Quand le premier cas a-t-il été détecté en France ?".

5.3.2 Limites et perspectives d'enrichissement du corpus TimeTank

Bien que le corpus TimeTank offre une ressource de valeur pour l'étude de la temporalité dans les publications scientifiques, certaines limites méritent d'être soulignées ainsi que des perspectives pour enrichir ce dernier.

TimeTank est actuellement restreint aux articles liés au SARS-COV et ses variants issus du corpus CORD-19. Il serait pertinent d'élargir les sources de données à divers domaines scientifiques afin d'obtenir une vision plus généralisable de l'usage des expressions temporelles à travers différents champs de recherche.

Le corpus TimeTank se concentre principalement sur les expressions temporelles absolues, il est à noter que nous omettons toute expression temporelle relative. Voici quelques exemples des expressions temporelles que nous ne prenons pas en compte dans ce corpus :

1. **Expressions temporelles relatives simples** : Des termes tels que “tomorrow”, “yesterday”, “today”, “now”, ou “soon” qui sont couramment utilisés pour décrire un moment par rapport au temps présent.

2. **Expressions qui expriment des durées relatives** : “for weeks”, “for months”, “for a few days”, etc.

3. **Expressions de fréquence** : “every day”, “once a year”, “rarely”, ou “occasionally” qui décrivent une occurrence régulière ou irrégulière d’un événement.

4. **Expressions liées à des événements** : Certaines expressions temporelles font référence à des moments spécifiques en relation avec des événements particuliers, tels que “after the election”, “before the ceremony”, “during the festival”, etc.

Il est important de reconnaître que si notre corpus couvre en majeure partie les expressions temporelles absolues, la complexité et la diversité des expressions temporelles dans la langue nécessitent une attention constante pour assurer une couverture exhaustive.

Par ailleurs, TimeTank gagnerait à être enrichi par des annotations linguistiques complémentaires (syntaxiques et sémantiques). Cet enrichissement offrirait une compréhension plus approfondie de l’usage des expressions temporelles. Le schéma TimeInfo permet un étiquetage syntaxique et sémantique avancé des expressions temporelles via des attributs additionnels. Par exemple, les attributs ‘startDuration’ et ‘endDuration’ fournissent les bornes précises d’un intervalle temporel. L’attribut ‘indicator’ recueille les éléments contextuels qui introduisent l’information temporelle. Les attributs ‘precision’ et ‘tempClue’ permettent de caractériser le niveau de précision d’expressions imprécises ("early", "mid", etc.). Enfin, l’attribut ‘valType’ distingue les valeurs temporelles réelles des valeurs estimées. Ces métadonnées syntaxico-sémantiques ouvrent des perspectives intéressantes pour des recherches d’information avancées prenant en compte les nuances des expressions temporelles. L’enrichissement syntaxique et sémantique de TimeTank grâce à ces attributs optionnels pourrait donc constituer une extension prometteuse du corpus.

Chapitre 6

Analyse de TimeBank et comparaison entre TimeML et TimeInfo

6.1 Analyse de TimeBank

Afin de comparer notre schéma d'annotation, nous utilisons des phrases annotés du corpus TimeBank[1] qui est un corpus annoté avec TimeML. TimeBank est une ressource précieuse pour les chercheurs travaillant sur des tâches impliquant un raisonnement temporel. Il est l'une des rares ressources de données textuelles annotées par des humains avec le schéma d'annotation TimeML. Dans cette partie, nous explorons le corpus annoté TimeBank et discutons de sa structure. Puis nous comparons des phrases annotées avec la balise TIMEX3 du schéma TimeML et notre schéma d'annotation TimeInfo.

TimeBank contient 183 documents, qui sont des articles de presse et d'autres types de textes annotés avec des informations temporelles. Chaque document est structuré selon le schéma TimeML, capturant une représentation détaillée des événements, des expressions temporelles et de leurs relations mutuelles.

Les types d'éléments d'annotations que nous trouvons dans le corpus TimeBank sont

[1] https://catalog.ldc.upenn.edu/LDC2006T08

structurés de la manière suivante :

- TIMEX3 : dates, heures, durées ;
- EVENT : éléments d'un texte qui marquent les événements sémantiques décrits par celui-ci ;
- MAKEINSTANCE : permet de créer une instance d'événement ou de signal temporel à partir d'un événement ou d'une expression temporelle balisée dans le texte ;
- SIGNAL : mots qui ont une fonction temporelle telle que "après", "pendant" et "quand" ;
- TLINK : relations temporelles entre les événements ;
- SLINK : relations de subordination entre les événements et/ou les expressions temporelles ;
- ALINK : relations entre les expressions temporelles et les événements dans un texte.

Dans notre projet de recherche, nous nous intéressons seulement à la partie temporelle du projet TimeML. En effet, la comparaison entre TimeInfo et TimeML se fait avec la balise TIMEX3, qui est utilisée pour l'annotation des expressions temporelles. Dans le corpus TimeBank, nous avons identifié 1414 occurrences de la balise TIMEX3. Les attributs ainsi que leurs fréquences sont représentés dans la figure 6.1. Ici, les résultats indiquent que les expressions temporelles de type 'DATE' sont les plus fréquentes, avec un total de 1 164 occurrences. Puis, en deuxième position, les expressions de type 'DURATION' sont représentées avec 175 occurrences. La fréquence des occurrences de type 'DURATION' est largement inférieure au type 'DATE'. Cela peut venir du fait que TIMEX3 n'est pas bien adapté pour l'annotation des durées, car ces dernières impliquent souvent l'utilisation d'une expression temporelle complexe qui peut contenir plusieurs unités temporelles.

Dans le tableau 6.1, nous présentons des exemples d'expressions temporelles que nous avons choisis aléatoirement, représentant les différents types d'annotations utilisés par TIMEX3. Dans ce tableau, chaque ligne correspond à une expression temporelle unique, identi-

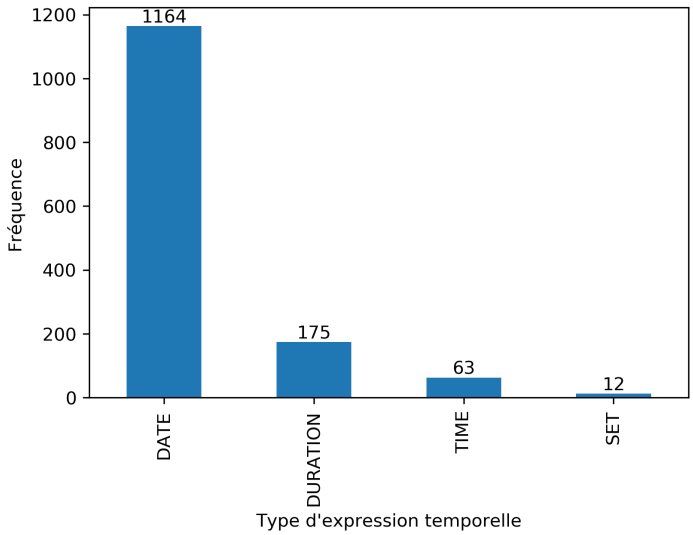


FIGURE 6.1 : Les occurrences des attributs de la balise TIMEX3

fiée par (tid). Les expressions sont classées en quatre catégories : DATE, DURATION, SET et TIME. La colonne "value" standardise ces expressions dans un format lisible par la machine, tandis que la colonne "text" présente l'expression temporelle telle qu'elle apparaît dans le corpus original. Ce tableau illustre la diversité des expressions temporelles et comment elles peuvent être standardisées pour une analyse systématique.

Dans l'analyse suivante, nous examinons chaque ligne du tableau en détail pour mieux comprendre comment ces expressions temporelles sont représentées et interprétées.

- **t42** : Cette ligne fait référence à une date spécifique, le 29 octobre 1989, exprimée dans le texte original comme "yesterday". Cela suggère que le texte a été écrit ou situé le 30 octobre 1989.
- **t128** : Ici, la date est simplement l'année 1978, exprimée de la même manière dans le texte original.
- **t68 (DATE)** : Cette ligne fait référence à l'année 1989, mais le texte original la décrit comme "fiscal 1989", ce qui pourrait indiquer une année fiscale plutôt qu'une année civile.

TABLE 6.1 : Exemples d'expressions temporelles dans TimeBank

tid	type	value	text
t42	DATE	1989-10-29	yesterday
t128	DATE	1978	1978
t68	DATE	1989	fiscal 1989
t361	DATE	PRESENT_REF	now
t32	DATE	1989-10-27	Friday
t126	DURATION	P9M	the first nine months
t68	DURATION	P9M	the nine months
t2012	DURATION	P9M	the first nine months
t327	DURATION	P14M	the next 14 months
t96	DURATION	PXD	A few days
t40	SET	XXXX-Q2	second
t157	SET	P1Y	a year
t107	SET	P1Y	each year
t190	SET	XXXX-WXX-1TNI	Monday
t2332	SET	P1M	every month
t100	TIME	1989-10-26	Later yesterday
t90	TIME	1989-10-27T24	midnight Friday
t54	TIME	1998-XX-XXTNI	the night
t25	TIME	PRESENT_REF	currently
t41	TIME	1998-02-13T14 :35 :00	02/13/1998 14 :35 :00

- **t361** : Cette ligne utilise “PRESENT_REF” pour indiquer une référence au moment présent, exprimée dans le texte original comme “now”.
- **t32** : Cette ligne fait référence à une date spécifique, le 27 octobre 1989, exprimée dans le texte original comme “Friday”. Cela suggère que le texte a été écrit ou situé peu de temps après cette date.
- **t126, t68, t2012, (DURATION)** : Ces trois lignes font toutes référence à une durée de neuf mois, exprimée dans le texte original comme “the first nine months” ou “the nine months”. Cela pourrait indiquer une période de temps spécifique dans une année fiscale ou civile.
- **t327** : Cette ligne fait référence à une durée de 14 mois, exprimée dans le texte original comme “the next 14 months”. Cela pourrait indiquer une projection ou une prévision pour une période future.
- **t96** : Cette ligne fait référence à une durée non spécifiée, exprimée dans le texte original comme “A few days”.
- **t157, t107** : Ces deux lignes font référence à une durée d’un an, exprimée dans le texte original comme “a year” et “each year”. Cela pourrait indiquer une période récurrente ou une durée spécifique.
- **t190** : Cette ligne fait référence à un ensemble de temps, spécifiquement un lundi non spécifié, exprimé dans le texte original comme “Monday”.
- **t2332** : Cette ligne fait référence à une durée d’un mois, exprimée dans le texte original comme “every month”. Cela suggère une récurrence mensuelle.
- **t100** : Cette ligne fait référence à une date et heure spécifiques, le 26 octobre 1989, exprimée dans le texte original comme “Later yesterday”. Cela suggère que le texte a été écrit ou situé le 27 octobre 1989.

- **t90** : Cette ligne fait référence à une date et heure spécifiques, minuit le 27 octobre 1989, exprimé dans le texte original comme “midnight Friday”. Cela suggère que le texte a été écrit ou situé peu de temps après cette date et heure.
- **t54** : Cette ligne fait référence à une heure non spécifiée en 1998, exprimée dans le texte original comme “the night”. Cela suggère une période de temps nocturne dans cette année.
- **t25** : Cette ligne utilise “PRESENT_REF” pour indiquer une référence au moment présent, exprimée dans le texte original comme “currently”.
- **t41** : Cette ligne fait référence à une date et heure spécifiques, le 13 février 1998 à 14h35, exprimées dans le texte original sous la forme “02/13/1998 14 :35 :00”.
- **t40** : Cette ligne fait référence à un ensemble de temps, qui a été calculé comme le deuxième trimestre d’une année non spécifiée, exprimé dans le texte original comme “second”.

À partir de ces exemples, nous pouvons observer que :

1. Les éléments `DURATION` décrivent des expressions qui ont toutes la forme “<nombre> + (days|months...)”, par exemple, “the nine months”, “a few days”. Les autres manières linguistiques pour exprimer une durée ne sont pas prises en compte. Par exemple, les expressions “de mars 2019 à septembre 2021” ou “entre midi et minuit le 19 octobre” expriment bien des durées mais ne seraient pas annotées comme `DURATION` par TIMEX3.
2. La différence entre `DATE` et `SET` n’est pas compréhensible hors du contexte du document. Il faut revenir aux expressions temporelles au sein de leur contexte afin de comprendre l’annotation de `SET`. Par exemple, “a year” est annoté avec `SET` alors que “1978” est annoté avec `DATE`. Le contexte permet de comprendre que “a year” fait en effet référence à un `SET` dans la phrase : *"Uh, in addition, white power racist music has become uh quite popular in certain sectors. Uh, there are over fifty thousand CDs* ***a year*** *sold with uh lyrics that are quite uh unbelievable."*

3. Certaines occurrences de `TIME` dénotent en réalité des durées (plusieurs heures dans une journée), par exemple, “Later yesterday” et “the night”. Or, toute expression temporelle qui contient une heure ou fait référence à des durées inférieures à une journée est annotée comme `TIME`.

L’exemple sur la figure 6.2 est un document issu du corpus TimeBank, annoté en utilisant le langage de balisage TimeML. Le document présenté ici est un article du Wall Street Journal daté du 25 octobre 1989, qui a été annoté pour mettre en évidence ces différentes dimensions temporelles.

Ici, nous allons mettre en exergue l’utilisation des balises TimeML : TIMEX3, EVENT, MAKEINSTANCE, TLINK et SLINK. Nous discuterons de la signification de ces balises et de la manière dont elles contribuent à la représentation structurée des informations temporelles dans le texte.

- **TIMEX3** : est utilisée pour annoter les expressions temporelles dans le texte. Par exemple :
  `<TIMEX3 tid="t11" type="DATE" value="1989-10-25" temporalFunction="false" functionInDocument="CREATION_TIME"> 10/25/89</TIMEX3>` indique que "10/25/89" est une date correspondant au 25 octobre 1989.

- **EVENT** : est utilisée pour annoter les événements dans le texte. Par exemple :
  `<EVENT eid="e2" class="REPORTING" stem="say">said</EVENT>` indique que "said" est un événement de type "REPORTING".

- **MAKEINSTANCE** : est utilisée pour créer des instances d’événements, qui peuvent ensuite être référencées dans les balises TLINK et SLINK.
  Par exemple : `<MAKEINSTANCE eventID="e2" eiid="ei1989" tense="PAST" aspect="NONE" polarity="POS" pos="VERB"/>` crée une instance de l’événement "e2" avec un certain nombre de propriétés, y compris le temps grammatical qui est ici passé.

```
<?xml version="1.0" ?>
<TimeML
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="http://timeml.org/timeMLdocs/TimeML_1.2.1.xsd">
 WSJ891025-0157
 = 891025
 891025-0157.
 Advanced Medical Makes Purchase
 <TIMEX3 tid="t11" type="DATE" value="1989-10-25" temporalFunction="false"
     functionInDocument="CREATION_TIME">10/25/89</TIMEX3>
 WALL STREET JOURNAL (J)
 AMA HENG
 TENDER OFFERS, MERGERS, ACQUISITIONS (TNM)
MEDICAL AND BIOTECHNOLOGY (MTC)
 SAN FRANCISCO
   Advanced Medical Technologies Inc. <EVENT eid="e2" class="REPORTING" stem="say">said</
       EVENT> it <EVENT eid="e3" class="OCCURRENCE" stem="purchase">purchased</EVENT> 93\%
       of a unit of Henley Group Inc.
   Advanced Medical <EVENT eid="e4" class="OCCURRENCE" stem="pay">paid</EVENT> $106 million
       in cash for its share in a unit of Henley's Fisher Scientific subsidiary.
The unit makes intravenous pumps used by hospitals and <EVENT eid="e9" class="OCCURRENCE"
    stem="have">had</EVENT> more than $110 million in <EVENT eid="e30" class="OCCURRENCE"
    stem="sale">sales</EVENT> <TIMEX3 tid="t31" type="DATE" value="1988" temporalFunction="
    true" functionInDocument="NONE" anchorTimeID="t11">last year</TIMEX3>, according to
    Advanced Medical.
<MAKEINSTANCE eventID="e2" eiid="ei1989" tense="PAST" aspect="NONE" polarity="POS" pos="VERB
    "/>
<MAKEINSTANCE eventID="e3" eiid="ei1990" tense="PAST" aspect="NONE" polarity="POS" pos="VERB
    "/>
<MAKEINSTANCE eventID="e4" eiid="ei1991" tense="PAST" aspect="NONE" polarity="POS" pos="VERB
    "/>
<MAKEINSTANCE eventID="e9" eiid="ei1992" tense="PAST" aspect="PERFECTIVE" polarity="POS" pos
    ="VERB"/>
<MAKEINSTANCE eventID="e30" eiid="ei1993" tense="NONE" aspect="NONE" polarity="POS" pos="
    NOUN"/>
<TLINK lid="l1" relType="SIMULTANEOUS" eventInstanceID="ei1992" relatedToEventInstance="
    ei1993" origin="USER"/>
<TLINK lid="l2" relType="BEFORE" eventInstanceID="ei1991" relatedToTime="t11" origin="USER"/
    >
<TLINK lid="l3" relType="BEFORE" eventInstanceID="ei1989" relatedToTime="t11" origin="USER"/
    >
<TLINK lid="l4" relType="BEFORE" eventInstanceID="ei1992" relatedToTime="t11" origin="USER"/
    >
<TLINK lid="l5" relType="DURING" eventInstanceID="ei1993" relatedToTime="t31" origin="USER"/
    >
<SLINK lid="l6" relType="EVIDENTIAL" eventInstanceID="ei1989" subordinatedEventInstance="
    ei1990"/>
</TimeML>
```

FIGURE 6.2 : Exemple d'un document du corpus TimeTank

- **TLINK** : est utilisée pour représenter les relations temporelles entre les événements et les expressions temporelles.
  Par exemple : `<TLINK lid="l1" relType="SIMULTANEOUS" eventInstanceID="ei1992" relatedToEventInstance="ei1993" origin="USER"/>`
  indique que les événements "ei1992" et "ei1993" se produisent simultanément.

- **SLINK** : est utilisée pour représenter les relations de subordination entre les événements. Par exemple :
  `<SLINK lid="l6" relType="EVIDENTIAL" eventInstanceID="ei1989" subordinatedEventInstance="ei1990"/>`
  indique que l'événement "ei1989" est une preuve de l'événement "ei1990".

L'un des avantages de l'utilisation de TimeML réside dans sa richesse, permettant d'annoter une variété d'informations temporelles, y compris les dates, les heures, les durées, les fréquences et les relations temporelles entre les événements. Cela permet une représentation détaillée de l'information temporelle dans les textes.

Cependant, l'utilisation de TimeML pour l'annotation temporelle présente également certains inconvénients. Tout d'abord, TimeML est un langage de balisage complexe avec de nombreux éléments et attributs, ce qui peut rendre son apprentissage et son utilisation difficiles pour les nouveaux utilisateurs. De plus, malgré la richesse expressive de TimeML, il peut toujours y avoir des ambiguïtés dans l'interprétation des annotations. Par exemple, une expression comme "la semaine prochaine" peut être interprétée de différentes manières en fonction du contexte. Ici, nous avons un exemple d'une expression temporelle relative, ce qui signifie qu'elle est interprétée par rapport à un certain point de référence dans le temps, souvent le moment présent. Cependant, le point de référence peut changer en fonction du contexte, ce qui peut conduire à différentes interprétations de l'expression. Par exemple, considérons la phrase "Nous organiserons une réunion la semaine prochaine". Si cette phrase est dite un lundi, "la semaine prochaine" se réfère généralement à la semaine commençant le lundi suivant. Ainsi, sans un point de référence clairement défini (c'est-à-dire une "date

d'ancrage"), il est impossible de déterminer avec certitude à quelle semaine précise l'expression fait référence. L'autre inconvénient de TimeML réside dans la difficulté de traiter des information temporelle complexes telles que les durées. Si nous prenons l'exemple "between the 12th and 17th March, 2020" l'annotation ressemblerait à ceci :

```
<TIMEX3 tid="t1" type="DATE" value="2020-03-12">the 12th March, 2020</
   TIMEX3>
<TIMEX3 tid="t2" type="DATE" value="2020-03-17">17th March, 2020</TIMEX3>
```

Afin d'annoter cette expression temporelle, il faut deux balises TIMEX3, une pour chaque date limite de l'intervalle. Cependant, il est important de noter que TimeML ne fournit pas de moyen direct d'exprimer la relation "between" entre ces deux dates. Pour faire le lien entre ces deux unités temporelles, nous devons utiliser des balises TLINK pour lier ces deux dates à un événement particulier, en indiquant que cet événement se produit pendant l'intervalle de temps spécifié. Cela nécessite une étape d'annotation supplémentaire et peut introduire des complications, en particulier si plusieurs événements se produisent dans le même intervalle de temps. De plus, l'annotation de ces expressions temporelles complexes peut être sujette aux erreurs et à l'incohérence, simplement en raison de leur complexité.

```
<TLINK eventInstanceID="e1" relatedToTime="t1" relType="AFTER"/>
<TLINK eventInstanceID="e1" relatedToTime="t2" relType="BEFORE"/>
```

Enfin, l'annotation de l'information temporelle en utilisant TimeML peut être une tâche coûteuse en temps et en ressources, en particulier pour les grands corpus de textes et c'est la raison pour laquelle le nombre de corpus annotés par des experts disponibles en libre accès est très limité.

6.2 Comparaison entre TIMEX3 et TimeInfo

Dans cette partie, nous allons extraire des exemples de TimeBank et les comparer avec TimeInfo. Nous allons voir comment TimeInfo simplifie l'annotation des expressions temporelles dans les textes.

TimeInfo présente plusieurs améliorations par rapport à l'approche de TimeML. Tout d'abord, elle permet une représentation explicite des intervalles de temps.

Exemple 1

Annotation avec TIMEX3 :

```
Mr. Lego the company foresees the need for a major boost in new-
    generation capability  <SIGNAL sid="s143">throughout</SIGNAL> <TIMEX3
    tid="t144" type="DURATION" value="199" temporalFunction="false"
    functionInDocument="NONE">the 1990s</TIMEX3>.
```

Annotation avec TimeInfo :

```
Mr. Lego the company foresees the need for a major boost in new-
    generation capability <TimeInfo interval="closed" granularity="decade"
     startDuration="1990" endDuration="1999" indicator="throughout">
    throughout the 1990s</TimeInfo>
```

Dans l'exemple ci-dessus, l'attribut 'interval' est utilisé dans TimeInfo pour indiquer que l'expression temporelle représente une durée fermée, c'est-à-dire une durée avec des dates de début et de fin précises. Avec TimeInfo, l'expression temporelle entière "throughout the 1990s" est traitée comme une seule entité, pour exprimer un événement qui s'est déroulé tout au long de la décennie. Avec TimeML, pour représenter cette même information il faut introduire la balise SIGNAL pour annoter l'élément "throughout". De plus, l'annotation selon le schéma TimeInfo précise explicitement que la décennie constitue une période délimitée, débutant en "1990" et se terminant en "1999".

Exemple 2

Annotation avec TIMEX3

```
Columbia also added $227.3 million to reserves for losses on the
   portfolio, increasing general reserves to $300 million, or about 6.7\%
    of the total portfolio, as of <TIMEX3 tid="t1991" type="DATE" value="
   1989-09-30" temporalFunction="true" functionInDocument="NONE"
   anchorTimeID="t163">Sept. 30</TIMEX3>.
```

Annotation avec TimeInfo

```
Columbia also added $227.3 million to reserves for losses on the
   portfolio, increasing general reserves to $300 million, or about 6.7\%
    of the total portfolio, <TimeInfo interval="left_open" granularity="
   day" endDuration="XXXX-09-30" indicator="as of">as of Sept. 30</
   TimeInfo>
```

Une différence notable entre TimeInfo et TimeML est que TimeInfo inclut un attribut `granularity` qui spécifie l'unité de temps utilisée pour mesurer la durée. Dans l'exmeple 2, la granularité est `day`, indiquant la plus petite unité de temps mentionnée. Cela offre une flexibilité supplémentaire pour représenter des durées dans différentes unités de temps, ce qui peut être particulièrement utile pour les expressions temporelles complexes. Aussi, TimeInfo utilise un attribut `duration` pour indiquer directement la longueur de la durée. Enfin, TimeInfo inclut des attributs `startDuration` et `endDuration` pour représenter directement les dates de début et de fin de la durée. Cela permet une représentation plus concise et moins ambiguë des intervalles de temps.

Dans l'exemple 2 ci-dessus, l'attribut `interval` de TimeInfo a une valeur `left_open` ce qui signale que l'expression temporelle représente un intervalle ouvert à gauche, impliquant que l'événement évoqué a débuté à une date non spécifiée, mais s'est achevé le '30 septembre'. TIMEX3 ne dispose pas d'un attribut qui indique cette information. De plus, avec l'attribut `indicator`, TimeInfo identifie les éléments linguistiques qui introduisent l'information temporelle. Dans l'exemple 2, l'indicateur est 'as of'.

Selon le schéma TimeInfo, si une unité temporelle n'est pas clairement identifiée dans le

texte de manière absolue, elle n'est pas annotée. En revanche, TIMEX3 prévoit le calcul des expressions temporelles relatives en se référant à l'information temporelle présente dans les méta-données du document :

```
Staff Reporter of The Wall Street Journal
<TIMEX3 tid="t163" type="DATE" value="1989-10-26" temporalFunction="false
    " functionInDocument="CREATION_TIME">10/26/89</TIMEX3>
```

Exemple 3

Annotation avec TIMEX3

```
But the group began to fall apart <SIGNAL sid="s167">in</SIGNAL> <TIMEX3
    tid="t143" type="DATE" value="1996" mod="MID" temporalFunction="false"
     functionInDocument="NONE">mid-1996</TIMEX3> after the defection of
    one of its top leaders, Ieng Sary.
```

Annotation avec TimeInfo

```
But the group began to fall apart <TimeInfo interval="closed" granularity
    ="year" precision="imprecise" tempClue="mid" value="1996" indicator="
    in">in mid-1996</TimeInfo> after the defection of one of its top
    leaders, Ieng Sary.
```

Dans l'exemple 3, TimeInfo propose une représentation plus détaillée que TimeML, en prenant en compte la précision, l'indice temporel, la granularité, et le type d'intervalle, rendant ainsi l'annotation plus fine.

Exemple 4

Annotation avec TimeInfo

```
Beyond these factors, viral spread was likely exacerbated further by the
   surge in domestic and international travel during the 40-day Lunar New
    Year (LNY) celebrations ( <TimeInfo interval="closed_duration"
   granularity="day" duration="40" startDuration="2020-01-10" endDuration
   ="2020-02-18" indicator="from ... to" precision="precise">from January
    10 th , 2020 to February 18 th , 2020</TimeInfo>) -the largest annual
    human migration in the world, comprised of hundreds of millions of
   people travelling across the country.
```

Annotation avec TIMEX3

```
Beyond these factors, viral spread was likely exacerbated further by the
   surge in domestic and international travel during the 40-day Lunar New
    Year (LNY) celebrations (from <TIMEX3 tid="t1" type="DATE" value="
   2020-01-10">January 10 th , 2020</TIMEX3> to <TIMEX3 tid="t2" type="
   DATE" value="2020-02-18">February 18 th , 2020</TIMEX3> the largest
   annual human migration in the world, comprised of hundreds of millions
    of people travelling across the country
```

Dans l'exemple 4, TIMEX3 propose deux annotations séparées pour les dates "January 10th, 2020" et "February 18th, 2020". En revanche, TimeInfo annote l'ensemble de la période comme une durée et capture également les deux unités temporelles qui la composent, tout en représentant la longueur de cette durée.

Exemple 5

Annotation avec TimeInfo

```
According to some sources, the first estimated date of reported COVID-19
    cases was <TimeInfo interval="closed" granularity="month" precision="
    precise" value="2019-12-XX" indicator="in" valType="estimated value">
    in December 2019</TimeInfo>.
```

Annotation avec TIMEX3

```
According to some sources, the first estimated date of reported COVID-19
    cases was <TIMEX3 tid="t1" type="DATE" value="2019-12"
    temporalFunction="false" functionInDocument="CREATION_TIME">in
    December 2019</TIMEX3>.
```

Dans l'exemple 5, l'attribut valType de TimeInfo a la valeur "estimated value", indiquant que la date "in December 2019" est une estimation. La distinction entre valeurs temporelles réelles et estimées a son importance pour un grand nombre d'applications. En particulier, cette information peut être utile pour l'étude de l'incertitude dans la littérature scientifique (voir par ex. [Atanassova and Rey, 2021]).

En utilisant le corpus TimeBank comme cas d'étude, l'analyse de TimeML souligne les caractéristiques et les défis de son utilisation dans l'annotation de données temporelles. TimeML, bien qu'étant un outil d'annotation temporelle riche et détaillé, présente des complexités qui peuvent poser des difficultés, surtout pour les nouveaux utilisateurs. Des ambiguïtés dans l'interprétation des annotations, comme l'exemple de "la semaine prochaine" ou "the next 14 months", illustrent des limites potentielles dans l'approche de TimeML, en particulier lorsque des références temporelles relatives sont utilisées sans points de référence.

En comparant TimeML avec TimeInfo, nous trouvons une simplification dans l'annotation des expressions temporelles. Cela pourrait potentiellement surmonter certaines des difficultés identifiées avec TimeML, telles que la complexité de son apprentissage et de son utilisation, ainsi que les défis liés à l'interprétation des expressions temporelles complexes ou ambiguës.

6.3 Remarques conclusives

La comparaison entre TimeInfo et TIMEX3, à travers les exemples cités ci-dessous, montre que :

- TimeInfo propose une représentation explicite des intervalles de temps, en un seul élément XML, alors que TIMEX3 utilise plusieurs éléments XML pour représenter les intervalles. L'interprétation des annotations dans TIMEX3 dans ces cas est plus difficile.
- TimeInfo inclut la granularité de l'information temporelle, cette information est utilisée pour le calcul des durées des intervalles fermés.
- TimeInfo capture les durées et leurs longueurs avec les attributs : `startDuration`, `endDuration` et `duration`, des données qui ne sont pas capturées dans les autres schémas d'annotation de l'information temporelle.
- TimeInfo propose une annotation simplifiée des expressions temporelles, ce qui peut faciliter son apprentissage par les utilisateurs novices.
- L'attribut `valType` de TimeInfo indique si la valeur de l'information temporelle énoncée est réelle ou estimée, une indication qui manque dans TIMEX3.

Cette comparaison montre la pertinence de notre schéma d'annotation, qui permet une représentation à la fois plus riche sémantiquement et plus intuitive des informations temporelles.

Une différence notable est le fait que TimeInfo est construit spécifiquement pour les textes scientifiques, et ainsi prend en compte les exemples issus des corpus d'articles scientifiques. TIMEX3, au contraire, est conçu pour tous types de textes, ce qui peut expliquer en partie le manque de précision quand il s'agit de représenter certaines informations, telles que les durées et les intervalles complexes.

Chapitre 7

TimeInfo : applications et perspectives

Le présent chapitre vise à démontrer comment notre schéma d'annotation TimeInfo, se positionne par rapport à d'autres approches à travers les applications et prototypes que nous avons développés.

7.1 Fine-Tuning d'un grand modèle de langage Llama-2-7b avec TimeTank

La formation et l'adaptation des grands modèles de langage (LLM) constituent des enjeux majeurs requérant des ressources conséquentes en calcul et en données. Toutefois, grâce aux progrès en intelligence artificielle, des solutions plus sophistiquées ont émergé pour faciliter ces opérations.

Des plateformes pour l'apprentissage automatique telles que PyTorch [Paszke et al., 2019] et TensorFlow [Abadi et al., 2016] offrent des interfaces dédiées à la manipulation et à l'adaptation des LLM. De plus, des méthodes innovantes de quantification, à l'instar de LoRa [Hu et al., 2021], ont été développées pour alléger et simplifier les LLM tout en préservant leur efficacité. Par ailleurs, des stratégies d'ajustement comme PEFT [Mangrulkar et al., 2022] optimisent les performances des LLM en affinant leurs

hyperparamètres. Nous avons utilisé les Transformers [Wolf et al., 2020] qui sont un outil puissant pour le traitement du langage naturel. Ces derniers peuvent être utilisés pour la génération de texte, la traduction automatique, dans les systèmes question/réponse, etc.

Afin d'exploiter le potentiel de TimeInfo, nous avons utilisé une partie du corpus TimeTank afin d'ajuster finement le grand modèle de langage Llama 2 pour extraire des expressions temporelles présentes dans des phrases et de les annoter avec les intervalles définis dans le schéma d'annotation TimeInfo.

Llama 2 est l'un des modèles de langage Open Source les plus avancés aujourd'hui. Il est disponible en plusieurs versions, allant de 7 milliards à 70 milliards de paramètres, il offre une flexibilité qui permet de répondre à divers besoins et applications. Llama 2 a été pré-entraîné sur des milliards de jetons ou tokens. Son architecture permet un ajustement fin sur des ensembles de données spécifiques, permettant aux chercheurs et aux développeurs de l'adapter à des tâches spécialisées [Touvron et al., 2023]. Pour son affinement, nous avons choisi le modèle 7B qui est composé de 7 milliards de paramètres. Ce modèle offre un équilibre entre performance et efficacité. Il est suffisamment grand pour capturer une grande partie de la complexité de la langue tout en maintenant une utilisation raisonnable des ressources informatiques.

Le Fine-Tuning est une méthode qui consiste à entraîner davantage un modèle pré-entraîné afin de l'adapter à une tâche précise [Howard and Ruder, 2018]. Dans notre recherche, nous avons utilisé cette approche pour faire en sorte que le modèle Llama-2-7b permette la détection et l'annotation d'expressions temporelles. Ainsi, le modèle affiné avec une partie de TimeTank arrive à détecter les expressions temporelles et à annoter leur intervalle.

Avant de commencer le Fine-Tuning du modèle Llama-2-7b, nous avons adapté et préparé les données de TimeTank afin qu'elles soient compatibles avec la structure utilisée par le modèle Llama2. Pour commencer, nous avons choisi aléatoirement 800 phrases : 200 phrases pour chacun des différents intervalles de TimeInfo. Puis nous avons structuré les données au format JSON Lines (.jsonl) où chaque ligne comporte deux éléments principaux :

prompt : Une phrase contenant une expression temporelle.

response : L'expression temporelle extraite de la phrase, suivie de la catégorie de l'intervalle basée sur le schéma TimeInfo.

Outre la structure duale des données (prompt/response), un troisième élément a été ajouté pour l'entraînement du modèle : un message système. Ce message système a été intégré au jeu de données d'entraînement. Cette étape était nécessaire car nous avons choisi de faire du fine-tuning sur la version conversationnelle (chat) du modèle Llama-2 7B. Le message système que nous avons utilisé pour cette tâche est le suivant : 'Given a sentence, you will identify and label the temporal expression using TimeInfo.'

Exemples :

```
{"prompt" "For example, on Feb 12, 2020, National Health Commission of China recommended to make clinical diagnosis besides pathogenic diagnosis when there were not enough kits and facilities to perform viral nucleic acid tests." "response" "on Feb 12, 2020 | closed"}
```

```
{"prompt" "In this review, we searched for all articles published in various databases including PubMed, Scopus, Embase, Science Direct and Web of Science using MeSH-compliant keywords including COVID-19, Pregnancy, vertical transmission, Coronavirus 2019, SARS-CoV-2 and 2019-nCoV from December 2019 to March 11 2020 and then reviewed them." "response" "December 2019 to March 11 2020 | closed_duration"}
```

```
{"prompt" "Up to February 24, 2020, the total number of patients had risen sharply to 77,269 confirmed cases and 2,596 deaths cases (http:\/\/2019ncov.chinacdc.cn\/2019-nCoV\/)." "response" "Up to February 24, 2020 | left_open"}
```

```
{"prompt" "Since December 2019, a new coronavirus has been emerging in the city of Wuhan, the capital of Hubei province in China." "response" "Since December 2019 | right_open"}
```

Le protocole d'entraînement du modèle Llama-2 s'articule autour des étapes suivantes :

1. Initialisation du modèle et du tokenizer

   D'abord, il s'agit d'initialiser le modèle Llama-2-7b-hf et le tokenizer issus de la bibliothèque Transformers. Le Llama-2-7b-hf est un modèle pré-entraîné sur un vaste corpus composé de textes et de code. Le tokenizer assure diverses tâches de pré-traitement des données, notamment la tokenisation et leur conversion en représentations numériques.

2. Configuration des paramètres d'entraînement

   Cette étape englobe la définition des divers paramètres qui guideront l'entraînement, tels que le nombre d'epochs, la taille des lots (batch) d'entraînement pour chaque unité GPU, etc.

3. Réglage des paramètres de fine-tuning

   Il est essentiel de déterminer les paramètres de fine-tuning, influençant la manière dont le modèle est adapté au jeu de données spécifique. Ceci inclut notamment les hyperparamètres relatifs à la technique de quantification LoRa.

4. Instanciation de l'objet SFTTrainer

   La classe SFTTrainer facilite le fine-tuning supervisé des modèles de langage de grande taille. Elle prend en entrée le modèle Llama-2-7b-hf, les jeux de données d'entraînement et de validation, le tokenizer, ainsi que les paramètres d'entraînement.

5. Processus d'entraînement

   À cette étape, le modèle Llama-2-7b-hf est formé à l'aide du jeu de données d'entraînement. La progression de l'entraînement est constamment surveillée par le suivi de la perte et de la précision à des intervalles définis.

6. Sauvegarde du modèle

   Après l'achèvement de l'entraînement, le modèle est sauvegardé dans un répertoire dédié, simplifiant ainsi son déploiement ou sa distribution ultérieure.

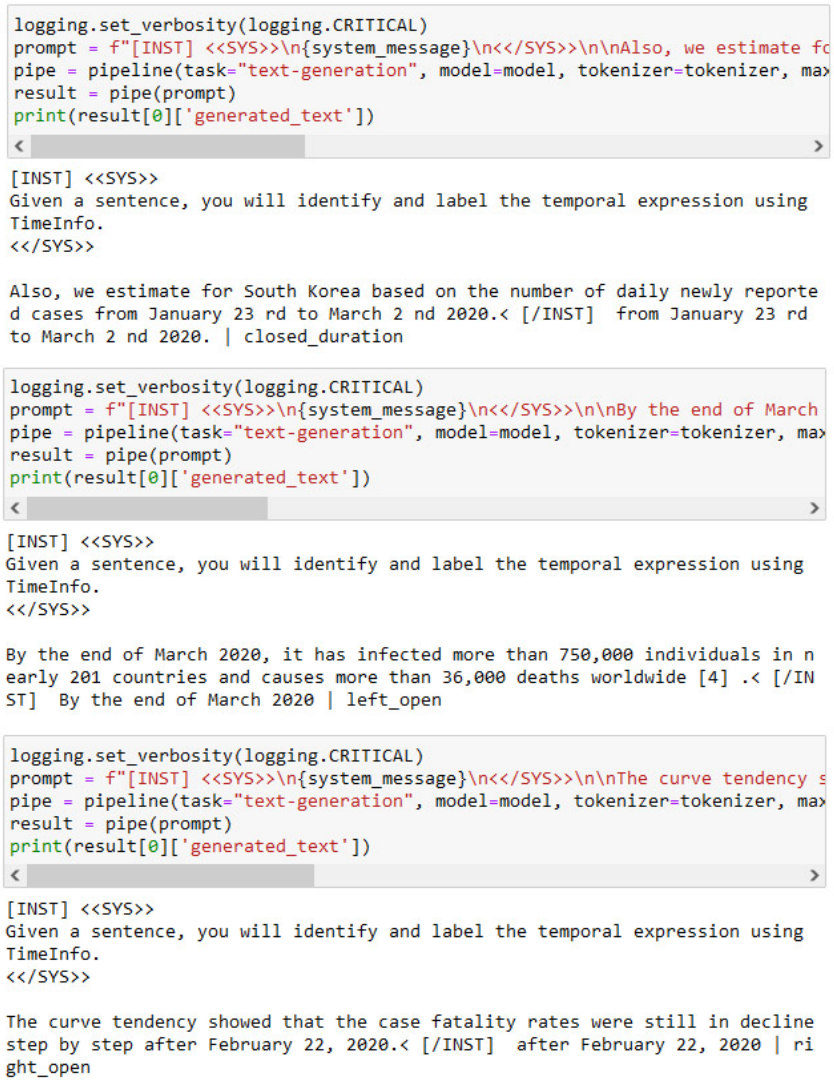
Dans le Tableau 7.1, nous présentons un récapitulatif des hyperparamètres utilisés lors du fine-tuning du modèle. Chaque hyperparamètre est accompagné de sa valeur respective ainsi que d'une brève description explicative. Ce tableau offre une vision synthétique des réglages et choix techniques effectués lors de cette étape d'ajustement.

TABLE 7.1 : Hyper-paramètres utilisés pour le fine-tuning

Hyperparamètre	Valeur	Description
lora_r	64	Le rayon utilisé pour la quantification.
lora_alpha	16	Le facteur d'échelle pour la quantification.
lora_dropout	0.1	Le taux de dropout pour la quantification.
use_4bit	True	Indique si la quantification en 4 bits doit être utilisée.
bnb_4bit_compute_dtype	float16	Le type de données de calcul pour la quantification 4 bits.
bnb_4bit_quant_type	nf4	Le type de quantification pour la quantification 4 bits.
use_nested_quant	False	Indique si la quantification imbriquée doit être utilisée.
output_dir	./results	Le répertoire où les résultats de l'entraînement seront enregistrés.
num_train_epochs	3	Le nombre d'epochs d'entraînement.
fp16	False	Indique si le calcul en demi-précision doit être utilisé.
bf16	False	Indique si le calcul en b float16 doit être utilisé.
per_device_train_batch_size	4	La taille du batch d'entraînement par GPU.
per_device_eval_batch_size	4	La taille du batch d'évaluation par GPU.
learning_rate	2e-4	indique la valeur de l'hyper-paramètre essentiel au sein des modèles d'apprentissage automatique, le taux d'apprentissage.
weight_decay	0.001	Le coefficient de décroissance du poids.
optim	paged_adamw_32bit	L'optimiseur à utiliser.
lr_scheduler_type	constant	Le type de planificateur d'apprentissage.
max_steps	-1	Le nombre maximum d'étapes d'entraînement.
warmup_ratio	0.03	Le rapport de réchauffement.
group_by_length	True	Indique si les données doivent être regroupées par longueur.
save_steps	25	Le nombre d'étapes d'entraînement entre les sauvegardes du modèle.
logging_steps	5	Le nombre d'étapes d'entraînement entre les enregistrements des logs.
max_seq_length	None	La longueur maximale de la séquence.
device_map	{"" : 0}	La mappage des GPU aux processeurs.

Le fine-tuning de notre modèle est fait avec un échantillon de 800 phrases. Malgré des résultats positifs (voir la figure 7.1), il est manifeste que la taille de cet échantillon reste modeste pour garantir une validation du modèle. Étant données les contraintes temporelles auxquelles nous avons été confrontés, il ne nous a pas été possible d'effectuer une évaluation pour le modèle fine-tuné. Cependant, dans le cadre de nos recherches futures, nous envisageons d'élargir significativement la taille de notre corpus d'entraînement. Conscients des défis computation-

nels que cela représente, nous prévoyons également d'allouer des ressources GPU adéquates, assurant la capacité de calcul nécessaire. Dès que ces éléments préparatoires seront réunis, nous pourrons alors entamer une évaluation de notre modèle.



```
logging.set_verbosity(logging.CRITICAL)
prompt = f"[INST] <<SYS>>\n{system_message}\n<</SYS>>\n\nAlso, we estimate fo
pipe = pipeline(task="text-generation", model=model, tokenizer=tokenizer, max
result = pipe(prompt)
print(result[0]['generated_text'])
```

```
[INST] <<SYS>>
Given a sentence, you will identify and label the temporal expression using
TimeInfo.
<</SYS>>

Also, we estimate for South Korea based on the number of daily newly reporte
d cases from January 23 rd to March 2 nd 2020.< [/INST]  from January 23 rd
to March 2 nd 2020. | closed_duration
```

```
logging.set_verbosity(logging.CRITICAL)
prompt = f"[INST] <<SYS>>\n{system_message}\n<</SYS>>\n\nBy the end of March
pipe = pipeline(task="text-generation", model=model, tokenizer=tokenizer, max
result = pipe(prompt)
print(result[0]['generated_text'])
```

```
[INST] <<SYS>>
Given a sentence, you will identify and label the temporal expression using
TimeInfo.
<</SYS>>

By the end of March 2020, it has infected more than 750,000 individuals in n
early 201 countries and causes more than 36,000 deaths worldwide [4] .< [/IN
ST]  By the end of March 2020 | left_open
```

```
logging.set_verbosity(logging.CRITICAL)
prompt = f"[INST] <<SYS>>\n{system_message}\n<</SYS>>\n\nThe curve tendency s
pipe = pipeline(task="text-generation", model=model, tokenizer=tokenizer, max
result = pipe(prompt)
print(result[0]['generated_text'])
```

```
[INST] <<SYS>>
Given a sentence, you will identify and label the temporal expression using
TimeInfo.
<</SYS>>

The curve tendency showed that the case fatality rates were still in decline
step by step after February 22, 2020.< [/INST]  after February 22, 2020 | ri
ght_open
```

FIGURE 7.1 : Expressions temporelles annotées avec un grand modèle de langage

7.2 Moteur de recherche basé sur TimeInfo

Afin de montrer une application possible de l'annotation des informations temporelles, nous avons implémenté un prototype de moteur de recherche TimeInfo Search (voir l'interface dans la Figure 7.2). Une caractéristique distincte de ce moteur est sa capacité à effectuer des recherches basées sur les attributs spécifiques de TimeInfo. Les utilisateurs peuvent, par exemple, effectuer des recherches en fonction de l'attribut "`interval`", ce qui permet de distinguer différents types d'événements ou durées.

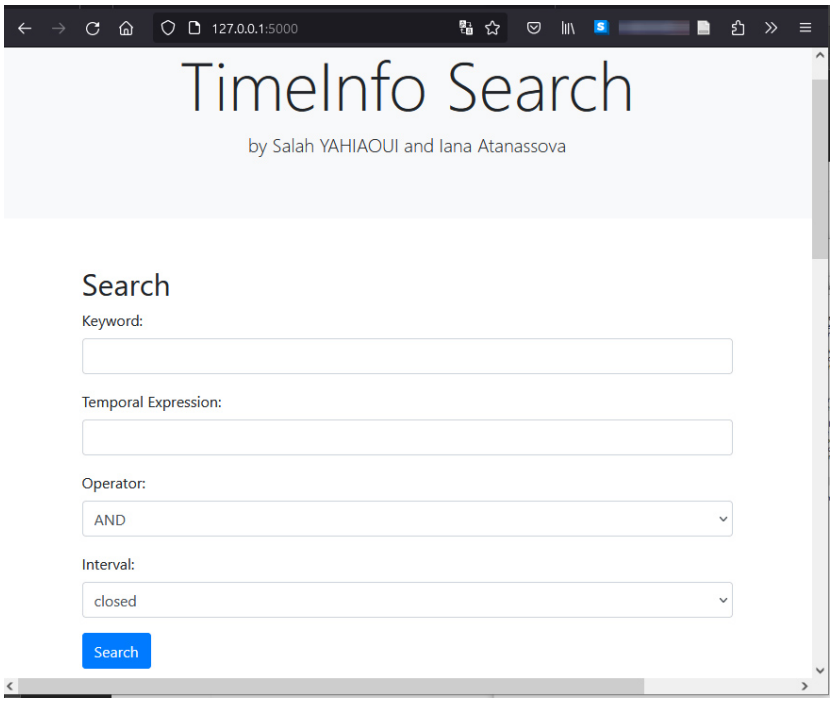


FIGURE 7.2 : Moteur de recherche : TimeInfo Search

À ce stade, TimeInfo Search est une application web prototype[1], utilisant le corpus TimeTank[2]. Actuellement, elle opère sur les mêmes jeux de données que TimeTank. Cet outil offre un accès à des phrases contenant au moins une expression temporelle annotée selon TimeInfo, permettant des recherches soit par mot-clé, soit par intervalle. Nous avons également

[1] http://timeinfo.pythonanywhere.com/
[2] https ://zenodo.org/records/8364409

intégré les opérateurs logiques ‘AND’ et ‘OR’ pour offrir davantage de flexibilité.

L’opérateur ‘OR’ permet d’afficher des résultats contenant l’un des termes ou les deux. Par exemple, une recherche pour *chat OR félin* renverrait des pages contenant soit "chat", soit "félin", soit les deux. L’opérateur ‘AND’ restreint les résultats à ceux qui contiennent tous les termes spécifiés. Ainsi, *chat AND sauvage* renverrait uniquement des pages contenant à la fois "chat" et "sauvage".

Dans TimeInfo Search, avec l’opérateur ‘OR’, l’utilisateur peut afficher des résultats contenant soit le mot-clé, soit l’expression temporelle, soit l’intervalle. À l’inverse, avec l’opérateur ‘AND‘, tous les éléments de la requête doivent être présents pour obtenir un résultat.

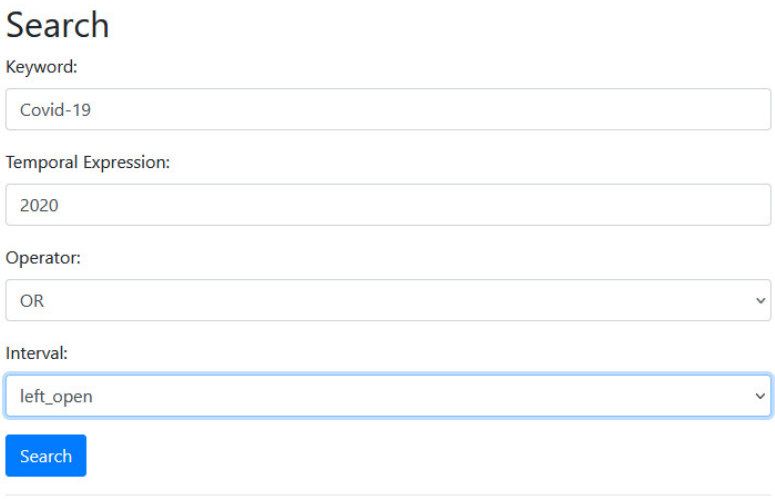


FIGURE 7.3 : Exemple de requête avec mot-clé, expression temporelle et opérateur OR

Par exemple, une requête utilisant l’opérateur ‘AND’ avec le terme ‘Covid-19’, l’année ‘2020’ dans l’expression temporelle et l’intervalle ‘`left_open`’ pour les expressions temporelles, fournirait les résultats présentés sur la figure 7.4

Cet outil permettrait aux épidémiologistes d’accéder à des indicateurs temporels granulaires sur la progression de la pandémie de COVID-19. Aussi, il faciliterait l’analyse, la synthèse et potentiellement la formulation d’hypothèses par les experts.

Article ID	Title	Sentence	Temporal Expression	Interval	Download
9e4b67ef6613c31882bb561d28770203da19771b	Effects of Chinese strategies for controlling the diffusion and deterioration of novel coronavirus-infected pneumonia in China	According to reports [4] , by March 1, 2020, more than 60 countries or areas had reported the confirmed patients with COVID-19, and more than 80,000 patients had been confirmed with COVID-19 worldwide.	by March 1, 2020	left_open	Download
9e4b67ef6613c31882bb561d28770203da19771b	Effects of Chinese strategies for controlling the diffusion and deterioration of novel coronavirus-infected pneumonia in China	As of March 6, 2020, the number of cumulative confirmed cases of COVID-19 has reached 100,000 worldwide, and some countries such as Korea, Japan, Italy, and Iran are also suffering from the rapid spread of COVID-19 [4, 7, 8] .	As of March 6, 2020	left_open	Download
0f8261c63cdeeb5c2bf9294eba2954e1ca71b244	Effect of non-pharmaceutical interventions for containing the COVID-19 outbreak in China	The median and interquatile range (blue) of estimates of COVID-19 cases are presented with reported cases (red) by date of illness onset as of February 13, 2020.	as of February 13, 2020	left_open	Download

FIGURE 7.4 : Résultats d'une requête avec mot-clé, expression temporelle et opérateur AND

Article ID	Title	Sentence	Temporal Expression	Interval	Download
317943dec06b88c1c62ef0ecd832f0d644b1d417	Evaluating new evidence in the early dynamics of the novel coronavirus COVID-19 outbreak in Wuhan, China with real time domestic traffic and potential asymptomatic transmissions	The novel coronavirus was first detected in Wuhan, China in December 2019 and three months later, 28 countries/regions have reported confirmed cases of COVID-19 infections, with a total of 43,101 confirmed cases globally	in December 2019	closed	Download
214ef8154bf31571fcb97fd44b8403df7e208e80	Development and Evaluation of an AI System for COVID-19 Diagnosis	The new coronavirus disease, now known as COVID-19 [1] , was first detected in Wuhan, China, in December 2019	in December 2019	closed	Download

FIGURE 7.5 : Résultats d'une requête avec mot-clé 'first detected' et l'intervalle 'closed'

Bien que le moteur de recherche TimeInfo Search présente des avancées notables dans la recherche d'informations temporelles, il est important de souligner qu'il s'agit actuellement d'un prototype. Il possède des limitations et des axes d'amélioration.

L'une des principales restrictions concerne l'exploitation partielle des attributs de TimeInfo. En effet, le moteur actuel se concentre principalement sur l'attribut "intervalle", laissant de côté d'autres attributs qui pourraient enrichir considérablement la recherche. Parmi ces attributs, la "valeur normalisée" des données temporelles représente un élément essentiel pour assurer une cohérence et une uniformité dans la recherche et l'interprétation des expressions temporelles.

De plus, TimeInfo offre la possibilité de distinguer entre les "valeurs estimées" et les "valeurs réelles". Cette distinction est fondamentale pour comprendre le degré de certitude ou d'approximation d'une expression temporelle. L'intégration de cette fonctionnalité permettrait aux utilisateurs d'affiner leurs recherches en fonction du degré de précision souhaité. Pour aller plus loin, le développement futur de TimeInfo Search devrait envisager d'exploiter pleinement ces attributs.

7.3 API basée sur TimeInfo

L'API TimeInfo a été élaborée en se basant sur l'architecture OpenAPI[3], une spécification largement adoptée pour la conception d'API. Adopter OpenAPI offre plusieurs avantages significatifs :

1. **Interopérabilité :** Étant une spécification reconnue, OpenAPI garantit que l'API TimeInfo peut interagir avec divers systèmes et technologies.
2. **Documentation automatique :** OpenAPI a la capacité de générer automatiquement une documentation détaillée et interactive. Cette documentation offre une interface utilisateur où les développeurs peuvent explorer et tester l'API.
3. **Facilité d'intégration :** Grâce à la nature standardisée d'OpenAPI, de nombreux outils et bibliothèques sont disponibles pour faciliter l'intégration dans différentes applications et plateformes.
4. **Évolutivité :** L'architecture permet des modifications et extensions aisées, garantissant ainsi que l'API peut évoluer selon les besoins futurs.

Concernant l'utilisation de l'API, un exemple de requête est le suivant :

```
GET /timeinfo/search?keyword=covid-19&interval=closed_duration
```

Cette requête interroge l'API sur des informations temporelles liées au "covid-19" avec un intervalle de type "`closed_duration`". En réponse, l'API fournira les données pertinentes correspondant aux critères de recherche.

L'API TimeInfo permet d'interroger et de récupérer des annotations temporelles, et offre une passerelle pour intégrer TimeInfo avec d'autres systèmes d'annotation, tels que TimeML. En particulier, l'API pourrait faciliter l'interaction entre TimeInfo et l'élément EVENT de TimeML.

[3] https://www.openapis.org/

À travers ce chapitre, nous avons exploré les diverses facettes et applications de TimeInfo. Nous avons vu comment TimeInfo peut être exploité pour améliorer la recherche, notamment avec le moteur de recherche TimeInfo Search. Bien qu'il soit encore à un stade prototypique, ce moteur montre déjà comment une meilleure annotation temporelle peut faciliter la recherche et l'analyse de vastes ensembles de données. L'adaptation du grand modèle de langage, Llama-2-7b, pour l'annotation avec TimeInfo démontre également le potentiel de cette méthodologie à s'intégrer avec les technologies modernes du traitement automatique du langage naturel. La mise à disposition de l'API TimeInfo souligne notre engagement à faciliter l'adoption de cette méthodologie par d'autres chercheurs et développeurs. Grâce à son architecture basée sur OpenAPI, elle offre une intégration facile et une grande portabilité.

Chapitre 8

Caractérisation et visualisation des données géospatiales de CORD-19

Le traitement des données temporelles présente certaines similarités avec l'extraction et l'identification des données spatiales. En effet, les données spatiales sont souvent introduites dans les textes par des entités nommées combinées avec un nombre limité de prépositions, similairement aux expressions temporelles. La recherche que nous avons présentée dans cette thèse, bien qu'elle se concentre sur les données temporelles, s'inscrit dans un projet plus vaste autour de l'extraction et l'exploitation des données spatio-temporelles issues d'articles scientifiques.

Dans ce chapitre, nous présentons une expérimentation et un travail préliminaire sur l'extraction et la catégorisation des données spatiales. Ce travail est complémentaire au traitement des données temporelles et permettrait d'ajouter une dimension spatiale à l'annotation des corpus scientifiques.

L'ambition de notre étude est d'engendrer de nouvelles connaissances en accumulant des informations issues de vastes ensembles de données textuelles. Les visualisations interactives et les outils développés suivant cette approche peuvent être utilisés par exemple, pour permettre d'identifier des études pertinentes en fonction d'une localisation géographique spécifique ou de comparer les résultats existants pour une région donnée. Ces perspectives pour-

raient guider les décisions concernant les lieux à privilégier pour de futures études, surveiller les recherches en cours et de manière plus large, appuyer la conception de systèmes de recherche d'informations scientifiques.

8.1 Méthode

Dans cette section, nous détaillons la procédure mise en œuvre pour traiter le texte, dans le but d'obtenir une visualisation pertinente de l'ensemble de données. Notre architecture de traitement se déploie en plusieurs étapes :

1. Extraction du contenu intégral des articles suivie d'une segmentation en phrases ;
2. Sélection des phrases mentionnant le "Covid-19" ou ses appellations similaires. Les étapes ultérieures se concentrent exclusivement sur ces phrases ;
3. Reconnaissance d'Entités Nommées : détection des localisations géographiques au sein des phrases. Seules les phrases incluant une référence géographique sont retenues pour les étapes suivantes ;
4. Géocodage pour déduire les coordonnées de latitude et de longitude associées à chaque phrase ;
5. Annotation sémantique pour classer et catégoriser les phrases, et par conséquent, les localisations géographiques associées.

L'ensemble des phrases ainsi annotées sert de base à une visualisation spatiale de l'ensemble de données.

8.2 Jeu de données

À l'instar de notre travail sur les données temporelles, nous avons exploré le COVID-19 Open Research Dataset (CORD-19)[1], une initiative de l'Allen Institute For AI, en collaboration avec d'autres acteurs, que nous avons récupéré le 16 avril 2020. Le corpus contient 59311 contributions scientifiques centrées sur le COVID-19, le SARS-CoV-2 et d'autres coronavirus.

8.2.1 Prétraitement et reconnaissance d'entités nommées

Nous avons identifié les phrases qui :

- Font mention du *Covid-19* ou de ses synonymes, comme *SARS-CoV-2, Coronavirus disease 2019, 2019-nCoV, SARS coronavirus 2*, etc ;
- Intègrent au moins une Entité Nommée de nature spatiale.

Pour le traitement, nous nous sommes appuyés sur SpaCy[2], une bibliothèque Python dédiée. Son outil de Reconnaissance d'Entités Nommées (NER) a été essentiel pour détecter les données spatiales et temporelles. Nous avons utilisé le modèle *en_core_web_lg* de SpaCy, entraîné sur le jeu de données OntoNotes[3], affichant une fiabilité de 86.74 %. Grâce à cette démarche, notre corpus final regroupe 15,016 phrases extraites de 2,766 articles.

8.2.2 Géocodage

Le géocodage désigne la conversion d'une mention de lieu en coordonnées géographiques, c'est-à-dire en latitude et longitude. Ce processus nous permet de déterminer les coordonnées géographiques à partir d'informations textuelles, telles que le nom d'une ville, une adresse ou encore un pays [Goldberg et al., 2007].

[1] https://www.kaggle.com/allen-institute-for-ai/CORD-19-research-challenge
[2] https://spacy.io
[3] https://catalog.ldc.upenn.edu/LDC2013T19

Afin de géocoder chaque phrase contenant des données spatiales, nous avons utilisé GeoPy[4], une bibliothèque Python qui s'appuie sur diverses API de services de géolocalisation, comme Google Maps, Azure Maps ou Nominatim. Pour notre étude, nous avons choisi Nominatim[5], un moteur de recherche basé sur les données d'OpenStreetMap[6]. Les coordonnées géographiques ainsi obtenues ont servi à la création d'une carte visuelle de notre ensemble de données. Les phrases mentionnant plusieurs lieux ont été associées à plusieurs coordonnées et apparaissent donc à divers endroits sur la carte.

Le processus de géocodage [Maisonobe et al., 2019] n'est pas exempt d'erreurs. Par exemple, l'utilisation de Nominatim peut parfois générer des coordonnées erronées pour des chaînes de caractères ambiguës correspondant à plusieurs lieux. De plus, l'outil de Reconnaissance d'Entités Nommées peut mal interpréter certaines mentions en tant que lieux alors qu'elles ont d'autres fonctions dans les phrases. À titre illustratif, dans la phrase *"Of these, 98.6% of patients had anti-SARS-CoV-2-IgG detected in sera, and 82.0% had anti-SARS-CoV-2-IgM detected in sera."*, le terme "sera" est incorrectement identifié comme une donnée spatiale. En réalité, il s'agit d'un terme médical, ce qui génère des coordonnées géographiques erronées pour cette phrase sur la carte. Bien que nous n'ayons pas quantifié ces erreurs, nos observations suggèrent qu'elles demeurent rares dans notre jeu de données et n'altèrent pas significativement la qualité de la visualisation.

[4] https://geopy.readthedocs.io/en/stable/
[5] http://nominatim.org/
[6] https://nominatim.openstreetmap.org/

8.2.3 Annotation sémantique

L'annotation et la catégorisation sémantique des données spatiales s'appuient sur notre travail antérieur portant sur la géolocalisation des maladies tropicales négligées [Yahiaoui and Atanassova, 2019]. Dans cette précédente étude, nous avions développé une méthodologie d'extraction et de traitement de données géospatiales à partir de publications scientifiques.

Ainsi, les phrases sélectionnées lors des étapes précédentes ont été annotées automatiquement avec les classes sémantiques suivantes :

1. Outbreak
2. Mortality
3. Confirmed cases
4. Research
5. Data about patients

Cet ensemble de classes a été élaboré afin de représenter divers types d'informations que l'on peut retrouver dans des phrases contenant des données géographiques. La Table 8.1 donne des exemples de phrases pour chaque classe. Certaines de ces classes peuvent être mises en relation avec des classes de l'Ontologie de Surveillance COVID-19[7]. Par exemple, "Détection de 2019-nCoV", "Exposition à la COVID-19" et "Enquête sur la COVID-19". Étant donné que d'autres ontologies émergeront probablement des recherches en cours, nous prévoyons d'étudier les possibles alignements d'ontologies dans nos travaux futurs.

Notre objectif est d'identifier les phrases pertinentes de l'ensemble de données et de les annoter en fonction de ces classes. Une phrase peut appartenir à aucune, une ou plusieurs classes. Pour cela, nous avons adopté une approche basée sur des règles, dans le but de produire un premier ensemble de données annotées qui pourra ensuite servir de données d'en-

[7] https://bioportal.bioontology.org/ontologies/COVID19/, Mars 2020

TABLE 8.1 : Catégorisation sémantique des phrases.

Catégorie Sémantique	Exemple
Outbreak	"As the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) pandemic hits Europe, it converges and interacts with three global crises that will make it spread even further : governance, economics, and migration."
	"We hereby describe the impact of the news of 2019-nCoV spread on the number of calls and response-related parameters of the major dispatch centre of the Emilia Romagna (ER) region and the measures taken in order to restore the service optimal efficiency."
Mortality	"Presently, Italy has one of the highest rate of SARS-CoV-2 infection in the world among large countries, with 143 cases per 100,000 people, the highest number of deaths and the highest mortality rate, 10.5% vs. an average value of 4.6%."
	"As of 22 February 2020, nearly 77,043 COVID-19 infections in humans have been confirmed in China, with at least 2,445 reported deaths."
Confirmed cases	"A novel coronavirus 2019-nCoV has been identified with the first confirmed patient cases in December 2019 in the city of Wuhan, capital of Hubei province, China. "
	"As of February 28, 2020 the COVID-19 outbreak has caused 78,961 confirmed cases (2791 deaths) across China, with the majority seen in Wuhan City, and 4691 cases (67 deaths) reported in the other 51 countries."
Research	"Our study suggests that both daily temperature and relative humidity influenced the occurrence of COVID-19 in Hubei province and in some other provinces. "
	"In this study, we used clinically representative large-scale datasets from three centers in Wuhan and two publicly available chest CT datasets to develop and evaluate an AI system for the diagnosis of COVID-19."
Data about	"Nasopharyngeal swabs were collected from patients presenting with symptoms of SARS-CoV-2 infection at multiple medical centers in Connecticut."
patients	"From January 21, 2019, when the first case of COVID-19 was identified, up to Feb.15, 2020, 2123 patients visited our Fever Clinic because of fever and/or respiratory symptoms, and 342 patients were confirmed to have pneumonia by CT scan or in a few cases by Chest X-ray."

traînement pour des algorithmes d'apprentissage automatique. Ces ensembles de règles sont illustrés dans l'extrait de code Python présenté en annexe 8.3.2.

La méthode pour la création des règles d'annotation s'appuie sur la théorie microsystémique, développée par Sylviane Cardey [Cardey, 2013] et de la méthode d'Exploration contextuelle, introduite par Jean-Pierre Desclés [Desclés, 1997]. Ils proposent, tous deux, des méthodes de construction et d'organisation de ressources linguistiques pour l'annotation de segments textuels selon des catégories sémantiques. Notre démarche s'inspire également de l'Étiquetage des Rôles Sémantiques (ERS). En Traitement Automatique des Langues, l'ERS est utilisé pour identifier un verbe, considéré comme prédicat, ainsi que ses arguments dans une phrase donnée [Bonial et al., 2010].

Par exemple, dans la phrase *"Data in this study showed that detected and confirmed cases with COVID-19 infection declined from the peak of 44 on January 8 to only 2 on January 19, 2020, suggesting that the epidemic was likely under control."*, le prédicat est *"showed"* et son argument est *"data in this study"*. Ce couple prédicat-argument, avec leurs expressions synonymes possibles, peut être identifié dans les phrases pour les étiqueter avec la classe "Research". Ainsi, pour chaque classe, nous avons élaboré un ensemble de règles sémantiques qui ciblent le prédicat et ses arguments. Ces règles servent de repères linguistiques pour attribuer une classe à la phrase.

8.3 Résultats

Dans le tableau 8.2, nous présentons le nombre de phrases annotées pour chaque classe. Au total, 7054 phrases ont été annotées, ce qui représente environ 47% de l'ensemble de données.

TABLE 8.2 : Phrases annotées

Classe	Nombre de phrases
Outbreak	2401
Mortality	692
Confirmed cases	2054
Research	2063
Data about patients	2081
Single annotation	5119
Multiple annotations	1935
Total number of sentences	7054

8.3.1 Évaluation de l'annotation sémantique

Nous avons procédé à une évaluation de l'annotation sémantique sur un échantillon de 200 phrases, sélectionnées aléatoirement dans notre ensemble de données. Chaque phrase a été manuellement annotée par deux experts, puis leurs annotations ont été comparées. Le taux de concordance entre les deux annotateurs était de 79,5%. Les phrases pour lesquelles il y avait un désaccord ont été ensuite examinées par un troisième expert afin de définir l'annotation finale.

Les annotations manuelles ont été comparées aux résultats produits par le système pour ces 200 phrases. Nous présentons les résultats dans la matrice de confusion du tableau 8.3. Pour les phrases comportant plusieurs annotations, chaque annotation est traitée comme une unité distincte dans cette matrice. Nous avons ensuite calculé la Précision, le Rappel et la mesure F-1 pour chaque classe. Les résultats sont présentés dans le tableau 8.4.

La classe "Outbreak" présente les scores les plus bas en termes de précision et de rappel, ce qui indique que les règles associées à cette classe nécessitent des améliorations. Pour les

TABLE 8.3 : Matrice de confusion des annotations

		Annotations du système					
		Outbreak	Mortality	C. cases	Research	Data	pas d'annotation
A. manuelle	Outbreak	**21**					9
	Mortality		**18**				2
	C. cases	1		**23**	1	3	3
	Research	1	1	1	**32**		6
	Data			1	1	**18**	2
	pas d'annotation	11	2	7	6	7	**63**

TABLE 8.4 : Évaluation des annotations

Class	Précision	Rappel	F-1
Outbreak	61.76 %	72.41 %	66.67 %
Mortality	85.71 %	90.00 %	87.80 %
Conf. cases	71.88 %	74.19 %	73.02 %
Research	80.00 %	78.05 %	79.01 %
Data	64.29 %	81.82 %	72.00 %
Pas d'annotation	74.12 %	65.63 %	69.61 %

autres classes, nous obtenons une mesure F-1 supérieure à 72%.

8.3.2 Visualisation spatiale de l'ensemble de données

L'ensemble des phrases annotées a été utilisé pour produire une visualisation spatiale interactive : l'utilisateur peut cliquer sur un lieu donné pour afficher des informations, telles que l'identifiant de l'article, la phrase et sa catégorie sémantique. Grâce à ces informations, l'utilisateur peut facilement accéder à l'article scientifique d'où provient une phrase et une géolocalisation particulière. Nous avons représenté plus de 3 200 phrases pour produire cette carte interactive, en utilisant Folium[8]. Folium est une bibliothèque Python qui utilise la plateforme Leaflet.js[9] pour visualiser des données sur une carte.

Les figures 8.1 et 8.2 montrent des exemples de la vue globale et une phrase affichée dans

[8] https://github.com/python-visualization/folium
[9] https://leafletjs.com/

l'interface respectivement. L'interface est accessible en ligne sur http://tesniere.univ-fcomte.fr/Salah/map_covid19.html.

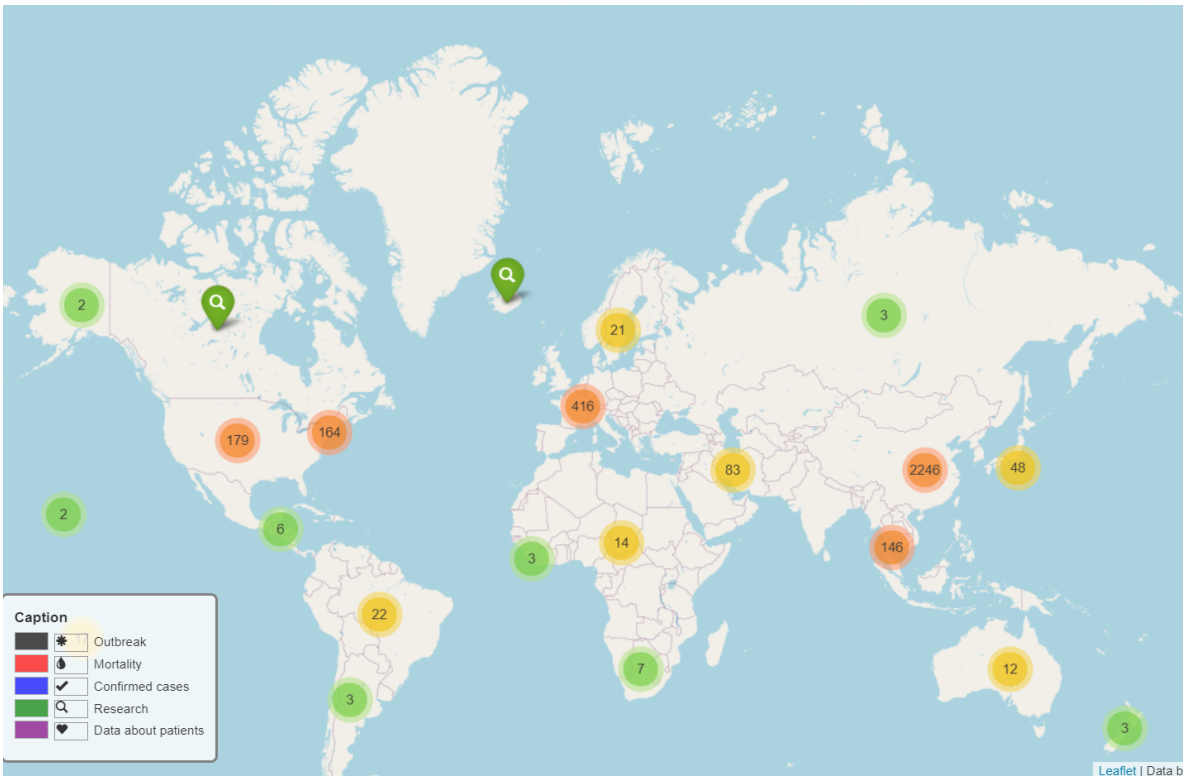


FIGURE 8.1 : Interface de visualisation spatiale produite en utilisant l'ensemble de données CORD-19

.

Cette forme de visualisation n'est pas conçue pour observer la progression des épidémies en temps réel dans les différents lieux géographiques. Au contraire, nous proposons d'étudier les articles de recherche. La visualisation produite reflète l'état actuel des connaissances que l'on peut trouver dans la recherche publiée sur le sujet. Les résultats de cette étude peuvent aider les chercheurs à accéder à des données sur un sujet particulier en fonction d'un emplacement géographique. De plus, notre carte peut être utilisée pour produire des informations complètes à l'intention des décideurs et pour déterminer de nouveaux sujets de recherche et politiques.

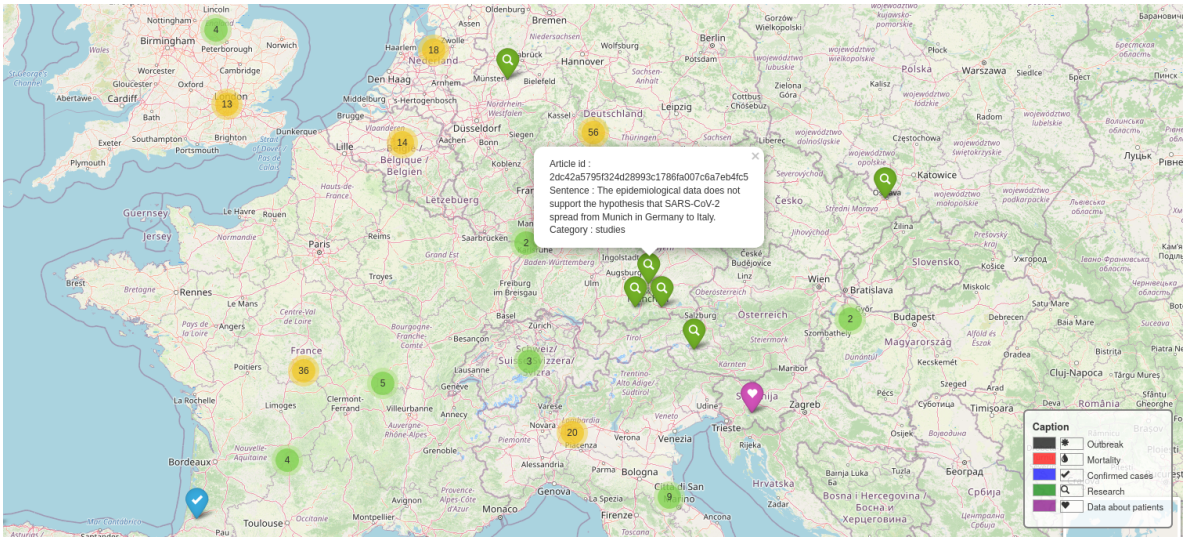


FIGURE 8.2 : Interface de visualisation spatiale des données CORD-19 avec un exemple de phrase affiché

Un autre axe prometteur de développement réside dans l'intersection de l'annotation sémantique géographique avec l'annotation temporelle. En conjuguant ces deux dimensions, il serait possible de suivre les données à la fois dans l'espace et le temps, offrant ainsi une représentation dynamique et multidimensionnelle des informations. Cette perspective, en reliant les lieux aux moments précis où des événements ou des découvertes ont eu lieu, pourrait enrichir considérablement la compréhension des tendances, des évolutions et des schémas émergents au sein de la littérature scientifique. Cette fusion des dimensions spatiales et temporelles pourrait ouvrir la voie à des analyses plus approfondies, permettant par exemple de suivre l'évolution de certains sujets de recherche ou tendances en fonction des régions ou des périodes.

Conclusion

Dans la présente thèse, nous avons proposé une nouvelle approche pour l'extraction et la catégorisation sémantique de l'information temporelle dans les textes scientifiques. Notre schéma d'annotation, TimeInfo, s'inspire en partie de standards existants, notamment TimeML. Cependant, notre approche s'est basée sur l'analyse des occurrences des expressions temporelles dans un corpus d'articles scientifiques, et la construction du schéma TimeInfo a été guidée par le besoin de rendre compte, le plus précisément possible, des différents aspects des informations temporelles. TimeInfo permet la représentation des expressions temporelles sous forme d'intervalles, qu'ils soient fermés ou ouverts. L'annotation rend compte également de la granularité des unités temporelles utilisées et permet de traiter les cas d'expressions temporelles complexes, notamment en prenant en compte leurs contextes. Les informations encodées par les divers attributs incluent les locutions introduisant les informations temporelles et les éléments linguistiques qui permettent de différencier une information temporelle estimée d'une information temporelle réelle.

Dans notre travail, nous avons montré que les schémas d'annotation de l'information temporelle, qui existent actuellement sous forme de standards, tels que TIMEX3, présentent plusieurs insuffisances quand il s'agit de rendre compte de toutes les propriétés sémantiques des expressions linguistiques. En même temps, leur appropriation par les annotateurs humains peut se révéler difficile, en raison de leurs spécificités et leur complexité intrinsèque. De plus, leur implémentation à l'aide d'outils informatiques n'est pas toujours triviale, ce qui peut engendrer des difficultés dans leur application automatisée. Au contraire, l'architecture de

TimeInfo permet de représenter les expressions temporelles complexes à l'aide d'un seul élément XML et de ses attributs. Cela rend son utilisation par les annotateurs humains plus facile et intuitive.

Notre projet de thèse vise également à démontrer la faisabilité d'une annotation automatique opérationnelle avec le schéma TimeInfo, en s'appuyant sur des ensembles de règles linguistiques. Nous avons proposé un ensemble de règles destinées, dans un premier temps, à détecter les expressions temporelles absolues, et ce, sans recourir aux outils de Reconnaissance d'Entités Nommées (REN) externes. Contrairement à ces outils, nos règles peuvent être modifiées pour les besoins spécifiques de chaque domaine d'application selon les corpus traités. Dans une seconde étape, nous avons élaboré un ensemble de règles linguistiques dédiées à la catégorisation des expressions temporelles, conformément aux intervalles définis par notre schéma TimeInfo. Ces règles linguistiques prennent en compte non seulement la nature propre de l'expression temporelle, mais également les locutions adverbiales, les adjectifs, les prépositions et autres éléments du contexte qui participent à la catégorisation des expressions temporelles. Les ensembles de règles linguistiques ont été implémentés dans un programme permettant d'extraire et d'annoter les expressions temporelles à partir de corpus scientifiques. Ce programme a été testé sur le corpus CORD-19.

La construction de corpus annotés d'une grande taille et d'une bonne qualité est un enjeu majeur pour la mise en place d'algorithmes d'apprentissage. Nous avons proposé un premier corpus TimeTank, annoté avec TimeInfo, et dont la qualité a été manuellement contrôlée. TimeTank, qui est disponible en accès libre[10], contient 1200 phrases annotées, reparties de manière égale sur les 4 types d'intervalles de TimeInfo. Dans l'objectif de proposer des pistes pour l'apprentissage de modèles basés sur TimeTank, nous l'avons utilisé pour affiner un Grand Modèle de Langage (LLM). Cette opération de "finetuning" visait principalement à spécialiser le LLM pour la détection et la catégorisation précises des expressions temporelles, conformément aux intervalles définis par TimeInfo. Une autre application de ce travail est le moteur de recherche sémantique que nous avons construit permettant d'exploiter la dimension temporelle des textes en s'appuyant sur les annotations. Bien que ces applications n'ont pas été évaluées dans cette thèse, notre objectif a été de démontrer avant tout les possibilités

[10] https://zenodo.org/record/8364409

d'applications qu'engendrent les annotations sémantiques proposées par TimeInfo. La prise en compte correcte de la sémantique des données temporelles dans les textes est un élément important de l'extraction d'informations. Enfin, nous avons également montré un prototype pour le traitement des informations spatiales et la cartographie des corpus scientifiques. L'intégration des deux types des traitements, des données temporelles et spatiales, permettrait la construction de nouveaux outils d'exploitation des publications, permettant la cartographie et la visualisation des thématiques scientifiques sous un angle spatio-temporel.

La priorité dans ce travail de thèse a été de proposer une nouvelle approche théorique pour l'annotation des informations temporelles. Ainsi, les implémentations informatiques comportent certaines limites, dont nous citerons ici quelques-unes. Les outils que nous avons développés, en l'état, sont des prototypes et ne sont pas encore prêts pour une mise en production à grande échelle. Notre outil d'annotation, par exemple, nécessite des améliorations significatives, notamment au niveau de son interface et API, afin d'être accessible pour les utilisateurs sans compétences en informatique. De même, le 'finetuning' du Grand Modèle de Langage a été réalisé sur un nombre limité de phrases. Pour garantir sa robustesse et son efficacité, une calibration sur un corpus plus vaste est indispensable. Quant au moteur de recherche, il demande non seulement un hébergement approprié, mais aussi une intégration avec diverses sources de données annotées pour optimiser sa pertinence et sa portée. Ces problématiques pourront être abordées dans nos futurs travaux de recherche.

Bibliographie

[Abadi et al., 2016] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., et al. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*.

[Al Hajjar et al., 2013] Al Hajjar, S., Memish, Z. A., and Mcintosh, K. (2013). Middle east respiratory syndrome coronavirus (MERS-CoV): a perpetual challenge. *Annals of Saudi medicine*, 33(5):427–436.

[Allen, 1983] Allen, J. F. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843.

[Anderson et al., 2004] Anderson, R. M., Fraser, C., Ghani, A. C., Donnelly, C. A., Riley, S., Ferguson, N. M., Leung, G. M., Lam, T. H., and Hedley, A. J. (2004). Epidemiology, transmission dynamics and control of SARS: the 2002–2003 epidemic. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, 359(1447):1091–1105.

[Araki et al., 2018] Araki, J., Mulaffer, L., Pandian, A., Yamakawa, Y., Oflazer, K., and Mitamura, T. (2018). Interoperable annotation of events and event relations across domains. In *Proceedings 14th Joint ACL-ISO Workshop on Interoperable Semantic Annotation*, pages 10–20.

[Atanassova, 2012] Atanassova, I. (2012). *Exploitation informatique des annotations sémantiques d'EXcom pour la recherche d'informations et la navigation*. PhD thesis, Université Paris 4.

[Atanassova, 2019] Atanassova, I. (2019). Beyond metadata: the new challenges in mining scientific papers. In *BIR@ ECIR*, pages 8–13.

[Atanassova et al., 2015] Atanassova, I., Bertin, M., and Kauppinen, T. (2015). Exploitation de données spatiales provenant d'articles scientifiques pour le suivi des maladies tropicales. In *Gestion et Analyse des données Spatiales et Temporelles (GAST), 15e conférence internationale sur l'extraction et la gestion des connaissances (EGC)*, Luxembourg.

[Atanassova et al., 2022] Atanassova, I., Bregnard, T., Abed, W. E., Isahara, H., and Cardey, S. (2022). Personal data in texts: Detection, annotation and governance. In *Panel in Computers, Privacy and Data Protection (CPDP) Conference*.

[Atanassova et al., 2021] Atanassova, I., Cardey-Greenfield, S., Madinier, H., and El Abed, W. (2021). Identification et gestion des données personnelles dans les textes. In *CiDE.22 : 22éme édition du Colloque International sur le Document Electronique Données Documents Connaissances : Perspectives de recherche et d'enseignement*, Paris, France.

[Atanassova and Rey, 2021] Atanassova, I. and Rey, F.-C. (2021). Categorising scientific uncertainty in papers. In *SciNLP 2021, 8 October 2021, 2nd Workshop on Natural Language Processing for Scientific Text*, pages https–scinlp.

[Baeza-Yates et al., 1999] Baeza-Yates, R., Ribeiro-Neto, B., et al. (1999). *Modern information retrieval*, volume 463. ACM press New York.

[Bertin and Atanassova, 2012] Bertin, M. and Atanassova, I. (2012). Semantic enrichment of scientific publications and metadata. *D-lib Magazine*, 18(7/8).

[Bethard et al., 2015] Bethard, S., Derczynski, L., Savova, G., Pustejovsky, J., and Verhagen, M. (2015). Semeval-2015 task 6: Clinical TempEval. In *proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 806–814.

[Biber et al., 1994] Biber, D., Conrad, S., and Reppen, R. (1994). Corpus-based approaches to issues in applied linguistics. *Applied linguistics*, 15(2):169–189.

[Bird, 2006] Bird, S. (2006). NLTK: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72.

[Bonial et al., 2010] Bonial, C., Babko-Malaya, O., Choi, J. D., Hwang, J., and Palmer, M. (2010). Propbank annotation guidelines. *Center for Computational Language and Education Research Institute of Cognitive Science University of Colorado at Boulder*.

[Brill, 1993] Brill, E. D. (1993). *A corpus-based approach to language learning*. University of Pennsylvania.

[Bzówka et al., 2020] Bzówka, M., Mitusińska, K., Raczyńska, A., Samol, A., Tuszyński, J. A., and Góra, A. (2020). Structural and evolutionary analysis indicate that the SARS-CoV-2 mpro is an inconvenient target for small-molecule inhibitors design. *bioRxiv*, pages 2020–02.

[Cardey, 2013] Cardey, S. (2013). *Modelling Language*. Natural Language Processing. John Benjamins Publishing Company.

[Cardey, 2022] Cardey, S. (2022). Semantic formal representation using indicants. In *Languages Analysis, Comparison and Generation Systems, Models and Applications, Homage to Peter GREENFIELD*, volume 40. Presses Universitaires de Franche-Comté (PUFC).

[Cardey and Greenfield, 2018] Cardey, S. and Greenfield, P. (2018). Scientific findings and their markers. In *Proceedings of the 4th Asia Pacific Corpus Linguistics Conference (APCLC)*, Takamatsu, Japan.

[Cardey et al., 2022] Cardey, S., Rey, F.-C., and Atanassova, I., editors (2022). *Languages Analysis, Comparison and Generation - Systems, Models and Applications. Homage to Peter Greenfield*. Bulag. Presses Universitaires de France-Comté.

[Carlson et al., 2004] Carlson, J. C., Byrd, B. D., and Omlin, F. X. (2004). Field assessments in Western Kenya link malaria vectors to environmentally disturbed habitats during the dry season. *BMC Public Health*, 4(1):1–7.

[Chang and Manning, 2012] Chang, A. X. and Manning, C. D. (2012). SUTime: A library for recognizing and normalizing time expressions. In *LREC*, volume 12, pages 3735–3740.

[Chen and Manning, 2014] Chen, D. and Manning, C. D. (2014). A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 740–750.

[Chinchor and Robinson, 1997] Chinchor, N. and Robinson, P. (1997). MUC-7 named entity task definition. In *Proceedings of the 7th Conference on Message Understanding*, volume 29, pages 1–21.

[Chinchor, 1998] Chinchor, N. A. (1998). Overview of MUC-7/MEC-2. Technical report, SCIENCE APPLICATIONS INTERNATIONAL CORP SAN DIEGO CA.

[Chiticariu et al., 2013] Chiticariu, L., Li, Y., and Reiss, F. (2013). Rule-based information extraction is dead! long live rule-based information extraction systems! In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 827–832.

[Chklovski and Pantel, 2004] Chklovski, T. and Pantel, P. (2004). Verbocean: Mining the web for fine-grained semantic verb relations. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 33–40.

[Chowdhary, 2020] Chowdhary, K. (2020). Natural language processing. *Fundamentals of artificial intelligence*, pages 603–649.

[Cowie and Lehnert, 1996] Cowie, J. and Lehnert, W. (1996). Information extraction. *Communications of the ACM*, 39(1):80–91.

[De Castilho et al., 2016] De Castilho, R. E., Mújdricza-Maydt, E., Yimam, S. M., Hartmann, S., Gurevych, I., Frank, A., and Biemann, C. (2016). A web-based tool for the integrated annotation of semantic and syntactic structures. In *Proceedings of the workshop on language technology resources and tools for digital humanities (LT4DH)*, pages 76–84.

[Derczynski and Gaizauskas, 2012] Derczynski, L. and Gaizauskas, R. (2012). A corpus-based study of temporal signals. *arXiv preprint arXiv:1203.5066*.

[Desclés, 1997] Desclés, J.-P. (1997). Systèmes d'exploration contextuelle. *Co-texte et calcul du sens*, pages 215–232.

[Desclés, 2006] Desclés, J.-P. (2006). Contextual exploration processing for discourse and automatic annotations of texts. In *Florida Artificial Intelligence Research Society (FLAIRS) Conference*, pages 281–284.

[Di Trani et al., 2007] Di Trani, L., Savarino, A., Campitelli, L., Norelli, S., Puzelli, S., D'Ostilio, D., Vignolo, E., Donatelli, I., and Cassone, A. (2007). Different pH requirements are associated with divergent inhibitory effects of chloroquine on human and avian influenza a viruses. *Virology journal*, 4(1):1–8.

[Dumais et al., 1998] Dumais, S., Platt, J., Heckerman, D., and Sahami, M. (1998). Inductive learning algorithms and representations for text categorization. In *Proceedings of the seventh international conference on Information and knowledge management*, pages 148–155.

[Edmonds, 2002] Edmonds, P. (2002). Senseval: The evaluation of word sense disambiguation systems. *ELRA newsletter*, 7(3):5–14.

[Edmonds and Cotton, 2001] Edmonds, P. and Cotton, S. (2001). Senseval-2: overview. In *Proceedings of SENSEVAL-2 Second International Workshop on Evaluating Word Sense Disambiguation Systems*, pages 1–5.

[Edmonds and Hirst, 2002] Edmonds, P. and Hirst, G. (2002). Near-synonymy and lexical choice. *Computational linguistics*, 28(2):105–144.

[El Abed et al., 2022] El Abed, W., Madinier, H., Bregnard, T., and Atanassova, I. (2022). Semantically-driven knowledge modelling for the business ecosystem. In *International Forum on Knowledge Asset Dynamics - IFKAD, Managing Knowledge for Sustainability*.

[Ellis and Post, 2004] Ellis, A. M. and Post, E. (2004). Population response to climate change: linear vs. non-linear modeling approaches. *BMC ecology*, 4(1):1–9.

[Ferro et al., 2003] Ferro, L., Gerber, L., Mani, I., Sundheim, B., and Wilson, G. (2003). Tides: 2003 standard for the annotation of temporal expressions. Technical report, MITRE CORP MCLEAN VA MCLEAN.

[Goldberg et al., 2007] Goldberg, D. W., Wilson, J. P., and Knoblock, C. A. (2007). From text to geographic coordinates: the current state of geocoding. *URISA-WASHINGTON DC-*, 19(1):33.

[Goldfarb, 1985] Goldfarb, C. F. (1985). The standard generalized markup language: Basic concepts. In *Offene Multifunktionale Büroarbeitsplätze und Bildschirmtext*, pages 132–140. Springer.

[Gries and Divjak, 2009] Gries, S. T. and Divjak, D. (2009). Behavioral profiles: a corpus-based approach to cognitive semantic analysis. *New directions in cognitive linguistics*, 57:75.

[Guisse and Atanassova, 2022] Guisse, A. and Atanassova, I. (2022). Noyau Informatique CripTex pour le Traitement de Données Personelles dans les Textes. *BULAG*, 40:597–612.

[Hirst, 1995] Hirst, G. (1995). Near-synonymy and the structure of lexical knowledge. In *AAAI Symposium on Representation and Acquisition of Lexical Knowledge: Polysemy, Ambiguity, and Generativity*, pages 51–56.

[Howard and Ruder, 2018] Howard, J. and Ruder, S. (2018). Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

[Hu et al., 2021] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. (2021). LoRa: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

[Huang and Lu, 2016] Huang, C.-C. and Lu, Z. (2016). Community challenges in biomedical text mining over 10 years: success, failure and the future. *Briefings in bioinformatics*, 17(1):132–144.

[Hunston, 2022] Hunston, S. (2022). *Corpora in applied linguistics*. Cambridge University Press.

[Jacobs and Hoste, 2022] Jacobs, G. and Hoste, V. (2022). Sentivent: enabling supervised information extraction of company-specific events in economic and financial news. *Language Resources and Evaluation*, 56(1):225–257.

[Jin et al., 2017] Jin, G., Atanassova, I., Soumana, I., and Cardey-Greenfield, S. (2017). Modèle pour un système multilingue d'analyse des opinons et de sentiment. In *Terminology & Ontology : Théories and applications (TOTh 2017)*, Chambéry, France. University of Savoie.

[Kang et al., 2020] Kang, S., Peng, W., Zhu, Y., Lu, S., Zhou, M., Lin, W., Wu, W., Huang, S., Jiang, L., Luo, X., et al. (2020). Recent progress in understanding 2019 novel coronavirus (SARS-CoV-2) associated with human respiratory disease: detection, mechanisms and treatment. *International journal of antimicrobial agents*, 55(5):105950.

[LeCun et al., 2015] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

[Liddy, 2001] Liddy, E. D. (2001). *Natural Language Processing*. Marcel Dekker, Inc., New York, 2nd edition.

[Lin, 2020] Lin, L. (2020). Solidarity with China as it holds the global front line during COVID-19 outbreak.

[Llorens et al., 2011] Llorens, H., Saquete, E., Navarro, B., Li, L., and He, Z. (2011). Data-driven approach based on semantic roles for recognizing temporal expressions and events in Chinese. In *Natural Language Processing and Information Systems: 16th International Conference on Applications of Natural Language to Information Systems, NLDB 2011, Alicante, Spain, June 28-30, 2011. Proceedings 16*, pages 88–99. Springer.

[Loper and Bird, 2002] Loper, E. and Bird, S. (2002). Nltk: The natural language toolkit. *arXiv preprint cs/0205028*.

[Maisonobe et al., 2019] Maisonobe, M., Jégou, L., Yakimovich, N., and Cabanac, G. (2019). Netscity: a geospatial application to analyse and map world scale production and collaboration data between cities. In *international conference on scientometrics and informetrics (ISSI 2019)*.

[Manfredi et al., 2014] Manfredi, G., Strötgen, J., Zell, J., and Gertz, M. (2014). Heideltime at eventi: Tuning italian resources and addressing TimeML's empty tags. *HeidelTime at EVENTI: Tuning Italian Resources and Addressing TimeML's Empty Tags*, pages 39–43.

[Mangrulkar et al., 2022] Mangrulkar, S., Gugger, S., Debut, L., Belkada, Y., and Paul, S. (2022). PEFT: State-of-the-art parameter-efficient fine-tuning methods. https://github.com/huggingface/peft.

[Mani et al., 2006] Mani, I., Verhagen, M., Wellner, B., Lee, C., and Pustejovsky, J. (2006). Machine learning of temporal relations. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 753–760.

[Mani et al., 2001] Mani, I., Wilson, G., Ferro, L., and Sundheim, B. M. (2001). Guidelines for annotating temporal information. In *Proceedings of the first international conference on Human language technology research*.

[Manning et al., 2014] Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.

[Màrquez et al., 2008] Màrquez, L., Carreras, X., Litkowski, K. C., and Stevenson, S. (2008). Special issue introduction: Semantic role labeling: An introduction to the special issue. *Computational Linguistics*, 34(2):145–159.

[Mehrabani et al., 2015] Mehrabani, M., Bangalore, S., and Stern, B. (2015). Personalized speech recognition for Internet of Things. In *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, pages 369–374. IEEE.

[Mihalcea et al., 2004] Mihalcea, R., Chklovski, T., and Kilgarriff, A. (2004). The senseval-3 English lexical sample task. In *Proceedings of SENSEVAL-3, the third international workshop on the evaluation of systems for the semantic analysis of text*, pages 25–28.

[Moorthy et al., 2020] Moorthy, V., Restrepo, A. M. H., Preziosi, M.-P., and Swaminathan, S. (2020). Data sharing for novel coronavirus (COVID-19). *Bulletin of the World Health Organization*, 98(3):150.

[Munafò, 2016] Munafò, M. (2016). Open science and research reproducibility. *ecancermedicalscience*, 10.

[Nasar et al., 2018] Nasar, Z., Jaffry, S. W., and Malik, M. K. (2018). Information extraction from scientific articles: a survey. *Scientometrics*, 117:1931–1990.

[Ng and Zelle, 1997] Ng, H. T. and Zelle, J. (1997). Corpus-based approaches to semantic interpretation in NLP. *AI magazine*, 18(4):45–45.

[Partalidou et al., 2019] Partalidou, E., Spyromitros-Xioufis, E., Doropoulos, S., Vologiannidis, S., and Diamantaras, K. (2019). Design and implementation of an open source greek pos tagger and entity recognizer using spacy. In *IEEE/WIC/ACM International Conference on Web Intelligence*, pages 337–341.

[Paszke et al., 2019] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

[Pöschl and Koop, 2008] Pöschl, U. and Koop, T. (2008). Interactive open access publishing and collaborative peer review for improved scientific communication and quality assurance. *Information services & use*, 28(2):105–107.

[Pustejovsky et al., 2003a] Pustejovsky, J., Castano, J. M., Ingria, R., Sauri, R., Gaizauskas, R. J., Setzer, A., Katz, G., and Radev, D. R. (2003a). TimeML: Robust specification of event and temporal expressions in text. *New directions in question answering*, 3:28–34.

[Pustejovsky et al., 2003b] Pustejovsky, J., Hanks, P., Sauri, R., See, A., Gaizauskas, R., Setzer, A., Radev, D., Sundheim, B., Day, D., Ferro, L., et al. (2003b). The TimeBank corpus. In *Corpus linguistics*, volume 2003, page 40. Lancaster, UK.

[Pustejovsky et al., 2005] Pustejovsky, J., Ingria, R., Saurí, R., Castaño, J., Moszkowicz, J., and Katz, G. (2005). The specification language timeml.

[Pustejovsky et al., 2010] Pustejovsky, J., Lee, K., Bunt, H., and Romary, L. (2010). Iso-timeml: An international standard for semantic annotation. In *LREC*, volume 10, pages 394–397.

[Ravin and Leacock, 2000] Ravin, Y. and Leacock, C. (2000). *Polysemy: Theoretical and computational approaches*. OUP Oxford.

[Saquete et al., 2006] Saquete, E., Munoz, R., and Martínez-Barco, P. (2006). Event ordering using TERSEO system. *Data & Knowledge Engineering*, 58(1):70–89.

[Saurí et al., 2006] Saurí, R., Littman, J., Knippen, B., Gaizauskas, R., Setzer, A., and Pustejovsky, J. (2006). TimeML annotation guidelines. *Version*, 1(1):31.

[Schilder and Habel, 2001] Schilder, F. and Habel, C. (2001). From temporal expressions to temporal information: Semantic tagging of news messages. In *Proceedings of the ACL 2001 workshop on temporal and spatial information processing*.

[Setzer, 2002] Setzer, A. (2002). *Temporal information in newswire articles: an annotation scheme and corpus study.* PhD thesis, University of Sheffield.

[Song and Chambers, 2014] Song, M. and Chambers, T. (2014). Text mining with the stanford corenlp. In *Measuring scholarly impact: Methods and practice*, pages 215–234. Springer.

[Sonnenburg et al., 2007] Sonnenburg, S., Braun, M. L., Ong, C. S., Bengio, S., Bottou, L., Holmes, G., LeCun, Y., Müller, K.-R., Pereira, F., Rasmussen, C. E., R228;tsch, G., Schölkopf, B., Smola, A., Vincent, P., Weston, J., and Williamson, R. (2007). The Need for Open Source Software in Machine Learning. *Journal of Machine Learning Research*, 8(81):2443–2466.

[Spellman et al., 2018] Spellman, B. A., Gilbert, E. A., and Corker, K. S. (2018). *Open Science*, pages 1–47. John Wiley Sons, Ltd.

[Stenetorp et al., 2012a] Stenetorp, P., Pyysalo, S., Topic, G., Ananiadou, S., and Aizawa, A. (2012a). Normalisation with the BRAT rapid annotation tool. *Semantic Mining in Biomedicine 2012*, page 87.

[Stenetorp et al., 2012b] Stenetorp, P., Pyysalo, S., Topić, G., Ohta, T., Ananiadou, S., and Tsujii, J. (2012b). Brat: a web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107.

[Strötgen et al., 2014] Strötgen, J., Bögel, T., Zell, J., Armiti, A., Van Canh, T., and Gertz, M. (2014). Extending heideltime for temporal expressions referring to historic dates. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2390–2397.

[Strötgen and Gertz, 2010] Strötgen, J. and Gertz, M. (2010). Heideltime: High quality rule-based extraction and normalization of temporal expressions. In *Proceedings of the 5th international workshop on semantic evaluation*, pages 321–324.

[Strötgen et al., 2013] Strötgen, J., Zell, J., and Gertz, M. (2013). Heideltime: Tuning English and developing spanish resources for TempEval-3. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 15–19.

[Suber, 2012] Suber, P. (2012). *Open access*. The MIT Press.

[Touvron et al., 2023] Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. (2023). Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

[Vasiliev, 2020] Vasiliev, Y. (2020). *Natural language processing with Python and spaCy: A practical introduction*. No Starch Press.

[Vega et al., 2004] Vega, V. B., Ruan, Y., Liu, J., Lee, W. H., Wei, C. L., Se-Thoe, S. Y., Tang, K. F., Zhang, T., Kolatkar, P. R., Ooi, E. E., et al. (2004). Mutational dynamics of the SARS coronavirus in cell culture and human populations isolated in 2003. *BMC Infectious Diseases*, 4(1):1–9.

[Velupillai et al., 2015] Velupillai, S., Mowery, D. L., Abdelrahman, S., Christensen, L., and Chapman, W. (2015). Blulab: Temporal information extraction for the 2015 clinical tempeval challenge. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 815–819.

[Verhagen et al., 2007] Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M., Katz, G., and Pustejovsky, J. (2007). SemEval-2007 task 15: TempEval temporal relation identification. In Agirre, E., Màrquez, L., and Wicentowski, R., editors, *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 75–80, Prague, Czech Republic. Association for Computational Linguistics.

[Verhagen et al., 2009] Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M., Moszkowicz, J., and Pustejovsky, J. (2009). The tempeval challenge: identifying temporal relations in text. *Language Resources and Evaluation*, 43(2):161–179.

[Vossen et al., 2016] Vossen, P., Agerri, R., Aldabe, I., Cybulska, A., van Erp, M., Fokkens, A., Laparra, E., Minard, A.-L., Aprosio, A. P., Rigau, G., et al. (2016). Newsreader: Using knowledge resources in a cross-lingual reading machine to generate more knowledge from massive streams of news. *Knowledge-Based Systems*, 110:60–85.

[Wang et al., 2020] Wang, L. L., Lo, K., Chandrasekhar, Y., Reas, R., Yang, J., Eide, D., Funk, K., Kinney, R., Liu, Z., Merrill, W., et al. (2020). Cord-19: The COVID-19 Open Research Dataset. *ArXiv*.

[Willinsky, 2006] Willinsky, J. (2006). *The access principle: The case for open access to research and scholarship*. Cambridge, Mass.: MIT Press.

[Wolf et al., 2019] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al. (2019). Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

[Wolf et al., 2020] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al. (2020). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45.

[Yahiaoui and Atanassova, 2019] Yahiaoui, S. and Atanassova, I. (2019). Géolocalisation des maladies tropicales négligées à l'aide de l'extraction et du traitement des données issues des articles scientifiques. Master's thesis, Université de Franche-Comté, CRIT EA 3224, Master LLCER, parcours Traitement Automatique des Langues (TAL).

[Yahiaoui and Atanassova, 2022] Yahiaoui, S. and Atanassova, I. (2022). Timeinfo: a semantic annotation framework for temporal information in scientific papers. In *Terminology & Ontology: Theories and applications (TOTH 2022)*, pages 161–174.

[Yahiaoui and Atanassova, 2023] Yahiaoui, S. and Atanassova, I. (2023). TimeTank: a corpus of sentences annotated with TimeInfo for temporal data. Dataset published in Zenodo.

[Yimam, 2019] Yimam, S. M. (2019). *Adaptive Approaches to Natural Language Processing in Annotation and Application*. PhD thesis, Staats-und Universitätsbibliothek Hamburg Carl von Ossietzky.

[Yimam et al., 2014] Yimam, S. M., Biemann, C., de Castilho, R. E., and Gurevych, I. (2014). Automatic annotation suggestions and custom annotation layers in webanno. In

Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pages 91–96.

[Yimam et al., 2015] Yimam, S. M., Biemann, C., Majnaric, L., Šabanović, Š., and Holzinger, A. (2015). Interactive and iterative annotation for biomedical entity recognition. In *International Conference on Brain Informatics and Health*, pages 347–357. Springer.

[Yimam et al., 2013] Yimam, S. M., Gurevych, I., de Castilho, R. E., and Biemann, C. (2013). Webanno: A flexible, web-based and visually supported system for distributed annotations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 1–6.

[Yuan et al., 2020] Yuan, M., Yin, W., Tao, Z., Tan, W., and Hu, Y. (2020). Association of radiologic findings with mortality of patients infected with 2019 novel coronavirus in Wuhan, China. *PloS one*, 15(3):e0230548.

[Zhang et al., 2020] Zhang, J., Litvinova, M., Liang, Y., Wang, Y., Wang, W., Zhao, S., Wu, Q., Merler, S., Viboud, C., Vespignani, A., et al. (2020). Age profile of susceptibility, mixing, and social distancing shape the dynamics of the novel coronavirus disease 2019 outbreak in china. *medrxiv*.

Annexe

Règles pour l'annotation sémantique des données spaciales

```
categories = []

# OUTBREAK
l_ourbreak = [  covid + r"\soutbreak|outbreak\s(began|started)",
                        r"(outbreak|spread|spreading|occurring|occurred)\
                            sof\s" + covid,
                        r"(outbreak|spread|spreading)\sin\s",
                        r"(outbreak|spread|spreading)\sof\spatients",
                        r"\boutbreak\soccurred",
                        r"\b(growth.*of|rise\b.*of)\b"+covid,
                        covid+r".*?(spread|spreading|occurring|occurred|
                            identified)"]


categories.append(category("outbreak", l_ourbreak))

# MORTALITY
deaths = r"\b([dD]eaths?|[Mm]ortality)\b"
l_mort = [r"(reported|occurred|causing|caused|causes?|confirmed|leading|
    leads?).*" + deaths,
                deaths + r"(counts?)?\sof\s" + covid,
                covid + r"\s" + deaths,
                r"diseased\spatients",
                r"mortality\srates?.*",
                r"total\sdeaths",
                r"(mortality|deaths?|died|dies?|dying)\s(observed|from|
                    for|of)\s.*"+covid,
                r"cases.*have died\s"]


categories.append(category("mortality", l_mort))
```

```
# CONFIRMED CASES
l_cc = [r"([cC]onfirmed|[iI]dentified|[Rr]eported|[d]eclared).*(cases?|
    infections?|patients?)",
            r"[Pp]atients\sinfected\swith.*" + covid + r".*\bwere\b",
            r"(cases?|infections?|patients?).*(were|was)\s([cC]onfirmed|[
                iI]dentified|[Rr]eported|[d]eclared)",
            r"(patients?\swith)\s(confirmed)?\s"+covid,
            r"cases?\s(identified|confirmed)",
            r"suspected cases?\b",
            r"number of cases of "+covid,
            r"tested positive for the virus"]


categories.append(category("confirmed cases", l_cc))


# STUDIES  -> RESEARCH
studies = r"\b(study|studies|researches|research|publications?|analys[ie]
    s|data|researchers?|observations?|results?|findings?|investigations?)\
    b"
l_studies = [ studies + r".*(obtained|proposed|conducted|conducts?|
    demonstrates?|suggests?|demonstrated|suggested|shows?|showed|focused|
    focuses|focus|presents?|found|describes?|described|indicates?|
    indicated|aimed|aims?|points?|investigates?|confrmed|confrms?|analy[sz
    ]ed|analyses?|provides?|provided|informed|informes?|exploring|explores
    ?|revealed|reported|reports?)"
            ,r"\b[Ww]e\b.*(identified|analyzed|searched|found|find|used|
                performed|show|showed|applied|conducted|developed|report)"
                ,
            r"\bto\b.*develop.*\bmodel\b",
            r"approaches.*answers to the questions?",
            r"survey conducted",
            r"([Tt]his|[Oo]ur|[Tt]hese) "+studies,
            r"[Tt]he "+studies+r" by\s"]
```

```
categories.append(category("research", l_studies))


# DATA ABOUT PATIENTS
l_data = [r"(patients?|cases?)\s(was|were|has|have|had|with|shows?|showed
    |tested)",
                r"([Pp]atiens?|[Cc]ases?).*(symptoms?|manifestation)",
                r"(symptoms?|diagnosis|diagnostics?|manifestation\sof).*(
                    patiens?|cases?)",
                r"patients?.*conditions?",
                r"medical waste of.*patients?"
     #           r"[Pp]atients? in.*\b(was|were)\b"
                ]


categories.append(category("data about patients", l_data))
```

LECLA
ÉCOLE DOCTORALE

Titre : Extraction et catégorisation d'informations temporelles de textes scientifiques

Mots clés : TimeInfo, extraction d'information, information temporelle, catégorisation sémantique, articles scientifiques, TAL

Résumé : Cette thèse aborde la problématique du traitement de corpus scientifiques, d'un point de vue linguistique, afin d'en extraire, catégoriser et agréger les informations spatio-temporelles pour produire de nouvelles représentations de l'information textuelle. Dans un premier temps, nous proposons le schéma d'annotation TimeInfo, qui permet de rendre compte de la sémantique des différentes expressions temporelles dans les textes scientifiques. Nous montrons l'apport de TimeInfo par rapport aux schémas d'annotation existants, notamment TimeML. Dans un deuxième temps, nous construisons des ensembles de règles linguistiques pour l'annotation automatique des corpus scientifiques avec TimeInfo. Nous traitons le corpus CORD-19 et produisons un nouveau corpus annoté, TimeTank. Enfin, nous proposons des applications autour de TimeInfo et abordons la problématique des informations spatiales, par une expérimentation sur leur annotation et cartographie.

Title: Extraction and categorization of temporal information from scientific texts

Keywords : TimeInfo, Information Extraction, temporal information, semantic classification, scientific papers, NLP

Abstract: This thesis addresses the problem of processing scientific corpora from a linguistic point of view in order to extract, categorise and aggregate spatio-temporal information in order to produce new representations of textual information. First, we propose the TimeInfo annotation scheme, which allows us to take into account the thematic nature of different temporal expressions in scientific texts. We show the contribution of TimeInfo compared to existing annotation schemes, in particular TimeML. Secondly, we construct sets of linguistic rules for the automatic annotation of scientific corpora with TimeInfo. We process the CORD-19 corpus and produce a new annotated corpus TimeTank. Finally, we propose applications based on TimeInfo and address the problem of spatial information by experimenting with its annotation and mapping.

UBFC
UNIVERSITÉ BOURGOGNE FRANCHE-COMTE

Université Bourgogne Franche-Comté
32, avenue de l'Observatoire
25000 Besancon