Apprentissage de représentation de graphes de connaissances et enrichissement de modèles de langue pré-entraînés par les graphes de connaissances : approches basées sur les modèles de distillation

Raphael Sourty

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par : *l'Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)*

Présentée et soutenue le *28/06/2023* **par :**

Raphaël Sourty

Apprentissage de représentation de graphes de connaissances et enrichissement de modèles de langue pré-entraînés par les graphes de connaissances : approches basées sur les modèles de distillation

---

JURY

Béatrice Daille	Professeure, Université Nantes	Rapporteure
Céline Hudelot	Professeure, CentraleSupélec, MISC	Rapporteure
Mohand Boughanem	Professeur, Université Toulouse 3	Examinateur
Benjamin Piwowarski	Chargé de Recherche, CNRS, LIP6	Examinateur
Lynda Tamine	Professeure, Université Toulouse 3	Directrice de thèse
José G. Moreno	MCF, Université Toulouse 3	Directeur de thèse
François-Paul Servant	Renault, IAA	Invité

École doctorale et spécialité :
MITT : Informatique et Télécommunications

Unité de Recherche :
Institut de Recherche en Informatique de Toulouse (UMR 5505)

Directeur(s) de Thèse :
Lynda TAMINE et *José G. MORENO*

Rapporteurs :
Béatrice DAILLE et *Céline HUDELOT*

Apprentissage de représentation de graphes de connaissances et enrichissement de modèles de langue pré-entraînés par les graphes de connaissances : approches basées sur les modèles de distillation

Raphael Sourty

28 Juin 2023

REMERCIEMENTS

Je tiens à exprimer ma gratitude sincère envers mes superviseurs de thèse, Lynda Tamine, professeure à l'université Paul Sabatier Toulouse III, José G. Moréno, maître de conférences à l'université Paul Sabatier Toulouse III, et François-Paul Servant, référent NLP et membre de l'équipe IAA chez Renault. Je suis extrêmement reconnaissant pour les connaissances et l'expérience qu'ils ont partagées avec moi, ainsi que pour les précieuses connaissances qu'ils m'ont transmises. Leurs conseils éclairés ont joué un rôle essentiel dans la réalisation de cette thèse.

Je souhaite exprimer ma profonde gratitude envers les membres du comité qui ont évalué ma thèse. Je tiens particulièrement à remercier Béatrice Daille, Professeure à l'Université de Nantes, Céline Hudelot, Professeure à CentraleSupélec et Directrice des recherches au MISC, Mohand Boughanem, Professeur à l'Université Toulouse 3, ainsi que Benjamin Piwowarski, Chargé de Recherche au CNRS LIP6.

Je suis profondément reconnaissant envers Renault d'avoir financé mes recherches doctorales au cours des trois dernières années. J'exprime mes sincères remerciements à Siham Essodaigui, Responsable de l'équipe IAA chez Renault, qui m'a accueilli au sein de son équipe et a veillé à ce que je bénéficie d'un environnement propice à la réussite de ma thèse.

Je souhaite exprimer ma profonde gratitude envers toutes les personnes qui ont contribué à rendre ma vie agréable et enrichissante au cours des quatre dernières années. Ils savent qui ils sont et je leur suis infiniment reconnaissant.

À Anne, Paul, Valérie et Emmanuel

RÉSUMÉ

Grâce aux avancées récentes en traitement du langage naturel et en intelligence artificielle, la capacité à traiter et à analyser efficacement les données pour générer des connaissances est devenue de plus en plus importante ces dernières années. Cela a permis de traiter un volume, qui continue de croître de façon significative, de données textuelles générées par les individus, les organisations et la société dans son ensemble.

Les bases de connaissances (KB) sont des structures qui encodent des informations sur les entités et les relations entre elles. Ils constituent un outil puissant qui permet de représenter les connaissances de manière structurée et formalisée, et de fournir une compréhension globale des concepts sous-jacents et de leurs relations. La capacité d'apprendre des représentations de graphes de connaissances a le potentiel de transformer le type de modèles de représentation sémantique qui sont actuellement en développement et qui seront développés dans les années à venir.

Les travaux menés dans cette thèse visent à explorer le concept de distillation des connaissances (KD) et, plus particulièrement, l'apprentissage mutuel pour l'apprentissage de représentations d'espace distincts et complémentaires. D'un côté nous avons la masse de données textuelles qui contient toute la richesse de l'expressivité d'une langue, et de l'autre, les connaissances structurées qui profitent d'une homogénéité dans sa représentation. Dans cette optique, nous explorons dans cette thèse ces deux sources d'information.

Notre première contribution est de proposer un nouveau cadre pour l'apprentissage d'entités et de relations sur des bases de connaissances multiples appelé KD-MKB. L'objectif clé de l'apprentissage de représentations multigraphes est d'améliorer les modèles d'entités et de relations avec différents contextes de graphes qui peuvent potentiellement faire le lien entre des contextes sémantiques distincts. Notre approche est basée sur le cadre théorique de la distillation des connaissances et de l'apprentissage mutuel. Elle permet un transfert de connaissances efficace entre les KBs tout en préservant la structure relationnelle de chaque graphe de connaissances. Nous formalisons l'inférence d'entités et de relations entre les bases de connaissances comme un objectif de distillation sur les distributions de probabilité postérieures à partir des connaissances alignées. Sur la base de ces résultats, nous proposons et formalisons un cadre de distillation coopératif dans

lequel un ensemble de modèles de KB sont appris conjointement en utilisant les connaissances de leur propre contexte et les softs étiquettes fournies par leurs pairs.

Notre deuxième contribution est une méthode permettant d'incorporer des informations riches sur les entités provenant de bases de connaissances dans des modèles de langage pré-entraînés (PLM). Nous proposons un cadre original, dans le contexte de PLMs, de distillation coopératif des connaissances pour aligner la tâche de pré-entraînement de modèles de langage masqués et l'objectif de prédiction de liens des modèles de représentation de KB. En exploitant les informations encodées dans les bases de connaissances et les modèles de langage pré-entraînés, notre approche offre une nouvelle direction de recherche pour améliorer la capacité d'un PLM à baser ses prédictions via des informations obtenues dans les bases de connaissances.

Dans nos deux contributions principales, nous avons utilisé des cadres expérimentales standards quand ils étaient disponibles (par exemple les ensembles de données basées sur Wordnet or Freebase) et nous avons proposé de nouveaux si nécessaire.

Finalement, nous concluons cette thèse et nous proposons de directions futures qui pourraient prendre les travaux à venir dans cette thématique de recherche.

ABSTRACT

With recent advances in natural language processing and artificial intelligence, the ability to efficiently process and analyze data to generate knowledge has become increasingly important in recent years. This has made it possible to process a volume, which continues to grow significantly, of textual data generated by individuals, organizations, and society as a whole.

Knowledge bases (KBs) are structures that encode information about entities and the relationships between them. They are a powerful tool for representing knowledge in a structured and formalized way, and for providing a comprehensive understanding of the underlying concepts and their relationships. The ability to learn knowledge graph representations has the potential to transform the type of semantic representation models that are currently under development and will be developed in the coming years.

The work in this thesis aims to explore the concept of knowledge distillation (KD) and, more specifically, mutual learning for learning distinct and complementary space representations. On the one hand we have the mass of textual data that contains all the richness of the expressiveness of a language, and on the other hand, structured knowledge that benefits from a homogeneity in its representation. In this perspective, we explore in this thesis these two sources of information.

Our first contribution is to propose a new framework for learning entities and relations on multiple knowledge bases called KD-MKB. The key objective of learning multigraph representations is to improve entity and relation models with different graph contexts that can potentially link distinct semantic contexts. Our approach is based on the theoretical framework of knowledge distillation and mutual learning. It allows efficient knowledge transfer between KBs while preserving the relational structure of each knowledge graph. We formalize the inference of entities and relationships between knowledge bases as a distillation objective on posterior probability distributions from aligned knowledge. Based on these results, we propose and formalize a cooperative distillation framework in which a set of KB models are jointly learned using knowledge from their own context and softs labels provided by their peers.

Our second contribution is a method for incorporating rich entity information from knowledge bases into pre-trained language models (PLMs). We propose an

original framework, in the context of PLMs, for cooperative knowledge distillation to align the task of pre-training masked language models with the goal of predicting links in KB representation models. By exploiting the information encoded in knowledge bases and pre-trained language models, our approach offers a new research direction to improve the ability of a PLM to base its predictions in information obtained via knowledge bases.

In our two main contributions, we used standard experimental frameworks when available (e.g., Wordnet or Freebase based datasets) and proposed new ones when necessary. Our results in extrinsic tasks proposed in recent benchmarks, as in the case of KILT, have shown improvements of our proposals when compared with state-of-the-art techniques.

Finally, we conclude this thesis and propose future directions that could be taken in this research topic.

PUBLICATIONS

Articles publiés dans des conférences internationales

1. **Raphaël Sourty**, Jose G. Moreno, François-Paul Servant et Lynda Tamine : Knowledge base embedding by cooperative knowledge distillation. *(Article long)* Dans : Proceedings of the 28th International Conference on Computational Linguistics, p. 5579–5590, Barcelona, Spain (Online), décembre 2020.
2. **Raphael Sourty**, Jose G. Moreno, Lynda Tamine et Francois-Paul Servant : Cherche : A new tool to rapidly implement pipelines in information retrieval. *(Article démo)* Dans : Proceedings of SIGIR 2022, 2022.
3. **Raphael Sourty**, Jose G. Moreno, Lynda Tamine et Francois-Paul Servant : Using CHERCHE to Empower Newcomers into Neural Information Retrieval. *(Article résumé)* Dans : Proceedings of CIRCLE 2022, 2022.

Articles publiés dans des conférences nationales

1. **Raphaël Sourty**, Jose G. Moreno, François-Paul Servant et Lynda Tamine : Enrichissement des modèles de langue pré-entraînés par la distillation mutuelle des connaissances. *(Article long)* Dans : CORIA-TALN 2023, 2023.

TABLE DES MATIÈRES

LISTE DES FIGURES

LISTE DES TABLEAUX

1

CONTEXTE ET CONTRIBUTION DE LA THÈSE

1 Contexte et problématique

1.1 *Contexte de la thèse*

Au cours des dernières années, la quantité de données textuelles générées par les individus, les organisations et la société dans son ensemble a connu une croissance significative (Hilbert et López, 2011), rendant la capacité à traiter et analyser le langage naturel critique. Le traitement automatique du langage naturel (TAL) est un domaine interdisciplinaire d'études visant à développer des méthodes et des modèles computationnels pour la compréhension et la modélisation du langage naturel (Manning et Schutze, 1999).

Les graphes de connaissances (KBs) sont des structures qui rassemblent des informations sur les entités et les relations entre elles. Ces entités peuvent inclure des personnes, des lieux et des concepts, et les relations peuvent inclure tout type d'association, comme la propriété, l'appartenance ou la causalité (Hogan *et al.*, 2021). Les KBs sont une source de premier choix pour obtenir une meilleure compréhension du langage naturel (Bast *et al.*, 2016).

Cette thèse traite de l'apprentissage de représentation sémantique des entités à partir de graphes de connaissances et de leur incorporation dans les modèles de langage pré-entraînés (PLMs) pour améliorer la compréhension du langage naturel. L'apprentissage de représentation sémantique de KBs consiste à encoder les entités dans un espace vectoriel qui est compréhensible pour les algorithmes (Bordes *et al.*, 2013).

Nous commençons par examiner les défis fondamentaux et les recherches menées pour repousser les limites de l'apprentissage des représentations de graphes de connaissances. Nous explorons ensuite l'apprentissage de représentation de mul-

tiples graphes de connaissances et les avantages potentiels de la combinaison de vues multiples d'une entité ou d'une relation donnée.

Ensuite, nous nous intéressons à la modélisation du langage naturel avec des modèles de langage tels que les Transformers. Nous examinons comment ces modèles peuvent être utilisés pour traiter le texte et comment l'utilisation de la distillation des connaissances permet de combiner l'espace de représentation des modèles de langage et des bases de connaissances.

2 Contributions

Les bases de connaissances (KBs) représentent une source de données de premier choix pour les tâches de traitement du langage naturel. Cette thèse propose un nouveau cadre pour l'apprentissage d'entités et de relations sur des KBs multiples, que nous appelons KD-MKB. Notre approche est basée sur le cadre théorique récent de la distillation des connaissances (Hinton *et al.*, 2015b), l'apprentissage mutuel (Zhang *et al.*, 2018), qui implique la distillation dans un cadre dynamique enseignant élève. Cette procédure permet aux modèles de KB d'"apprendre" les uns des autres et d'améliorer leurs performances dans la tâche de prédiction de liens (Bordes *et al.*, 2013). Nous formalisons notamment l'inférence d'entités et de relations entre les bases de connaissance comme un objectif de distillation sur les distributions de probabilité postérieures à partir des connaissances alignées. Sur la base de cette découverte, nous proposons et formalisons un cadre de distillation coopératif dans lequel un ensemble de modèles de KBs sont appris conjointement. Notre approche permet un transfert efficace des connaissances entre les KBs tout en préservant la structure relationnelle de chaque graphe de connaissances (Sourty *et al.*, 2020).

Dans cette thèse, nous proposons une procédure qui incorpore dans les modèles de langage des informations riches sur les entités à partir des KBs. Notre stratégie d'enrichissement des PLMs repose sur l'alignement de deux tâches existantes : la prédiction de liens (Bordes *et al.*, 2013) et le MLM (Devlin *et al.*, 2019). Nous tirons parti des corpus textuels mentionnant des entités pour estimer et transférer les probabilités d'entités d'une représentation KB à un MLM et vice versa, via la procédure d'apprentissage mutuelle (Zhang *et al.*, 2018). Nous soutenons que le corpus textuel et la base de connaissances représentent deux espaces distincts et complémentaires. Nous évaluons notre approche sur la tâche de slot-filling, qui fait référence à l'objectif d'extraction d'informations structurées à partir de textes non structurés. Nous montrons que les systèmes actuels de slot-filling basés sur les

PLMs bénéficient du cadre que nous avons défini, en particulier dans leur capacité à traiter les entités.

3 Organisation de la thèse

Le chapitre 2 traite des graphes de connaissances, incluant les principes fondamentaux du domaine et les ressources standards associées. Le chapitre 2 introduit l'apprentissage des représentations des entités et des relations dans les graphes de connaissances tels que les modèles géométriques, les modèles de décomposition en tenseurs, et les modèles neuronaux, ainsi que des modèles de représentation multi-KB. La troisième partie du chapitre aborde les procédures d'évaluation des modèles de KB et les ressources associées.

Le chapitre 3 introduit les techniques récentes de modélisation du langage naturel incluant les embeddings de mots et les modèles de langage profond. Ce chapitre se concentre aussi sur la distillation des connaissances, pour la tâche intrinsèque de modélisation du langage masqué, la distillation des prédictions dédiée à la tâche extrinsèque de classification, et la distillation des représentations. La troisième partie aborde l'apprentissage mutuel, incluant l'algorithme d'apprentissage mutuel et son application dans le cadre du traitement du langage naturel.

Le chapitre 4 analyse les méthodes utilisées pour injecter les informations des entités disponibles dans les graphes de connaissances au sein des modèles de langue. Dans ce travail, nous cherchons à comprendre les différentes procédures pour augmenter les modèles de langage avec des graphes de connaissances et à analyser les différents paradigmes pour représenter les entités et le langage dans un espace commun.

Nous présentons notre première contribution au domaine de l'apprentissage des représentations de graphes de connaissance dans le chapitre 5. Le chapitre 6 détaille notre modèle pré-entraîné de façon coopérative dans le cadre de l'augmentation des PLMs par la distillation des bases de connaissances.

Partie I

SYNTHÈSE DES TRAVAUX DE L'ÉTAT-DE-L'ART

2

APPRENTISSAGE DE LA REPRÉSENTATION DES CONNAISSANCES VIA LES GRAPHES DE CONNAISSANCES

Introduction

Un graphe de connaissances (KB) est une ressource qui rassemble et structure un ensemble d'entités. Pour représenter des faits, le KB connecte les entités entre elles via des relations dirigées. Il existe plusieurs formats disponibles pour représenter l'information dans une KB, cependant ils sont majoritairement basés sur le concept de triplet. Par exemple, un des formats est le standard Resource Description Framework (RDF). L'objectif du modèle de graphe RDF est d'uniformiser les ressources et faciliter le traitement des connaissances en organisant les faits sous la forme de triplets : (sujet, prédicat, objet). Le standard RDF a été popularisé par le Web sémantique (Berners-Lee *et al.*, 2001) dont l'ambition est de faciliter le traitement, le partage et la création des connaissances pour les applications et les utilisateurs du web.

Un KB est par conséquent une ressource qui permet d'organiser et de partager la connaissance avec les utilisateurs et les applications. Le KB exprime le fait que le jaguar vit en Amérique du Sud via une relation sémantique : (*jaguar, habite, Amérique du Sud*). Nous pouvons représenter le fait que le Jaguar est un félin via une relation de subsomption (*jaguar, est, félin*). La relation de subsomption exprime l'appartenance à une catégorie (hyponymie) : le jaguar appartient à la classe des félins.

Mettre à jour les KBs pour garantir la qualité de la connaissance représentée nécessite des efforts importants (Tang *et al.*, 2019a). L'engouement pour les KBs a révélé la nécessité d'automatiser leur construction et leur maintenance à partir de corpus de textes (Luan *et al.*, 2018). Dans la littérature, ces systèmes de population de KB sont généralement composés d'un système de résolution de coréférence d'en-

tités, un système de reconnaissance ou de désambiguïsassions d'entités et d'une méthode d'extraction ou de classification des relations (Ellis *et al.*, 2015).

Les KBs sont incomplets par nature (Rossi *et al.*, 2021), 71% des personnes décrites dans le KB Freebase n'ont pas de lieu de naissance et 75% d'entre elles n'ont pas de nationalité connue (Minervini *et al.*, 2015). Pour inférer des relations entre les entités d'un graphe, on retrouve des méthodes supervisées qui apprennent une représentation sémantique des entités et des relations en tenant compte de la structure du graphe. On distingue, parmi d'autres, dans l'état de l'art les modèles translationnels (Bordes *et al.*, 2013; Wang *et al.*, 2014a; Lin *et al.*, 2015a) et les modèles de correspondance sémantique (Nickel *et al.*, 2011; Yang *et al.*, 2015a; Trouillon *et al.*, 2016; Nickel *et al.*, 2016; Dettmers *et al.*, 2018; Sun *et al.*, 2019b). Ces modèles bénéficient de la représentation numérique des entités pour effectuer des opérations mathématiques tel que **Jaguar** $+$ **cousin** $\approx$ **Léopard** (modèles translationnels).

Les représentations sémantiques obtenues à partir d'une KBs sont utiles pour une variété d'applications dans le cadre du traitement du langage naturel : classification de relations (Peters *et al.*, 2019; Zhang *et al.*, 2019d; Yamada *et al.*, 2020; Wang *et al.*, 2021a; Poerner *et al.*, 2020), reconnaissance d'entités (Peters *et al.*, 2019; Zhang *et al.*, 2019d; Yamada *et al.*, 2020; Poerner *et al.*, 2020), désambiguïsation d'entités (Peters *et al.*, 2019; Poerner *et al.*, 2020), définition du type des entités (Peters *et al.*, 2019; Yamada *et al.*, 2020; Wang *et al.*, 2021a), compréhension générale du langage naturel (Zhang *et al.*, 2019d; Wang *et al.*, 2021a).

Dans le cadre de cette thèse, nous nous intéressons aux travaux et aux ressources permettant de construire des représentations sémantiques des entités et des relations. En particulier, nous étudierons les procédures permettant de mettre à jour une ou plusieurs bases de connaissances.

1 Graphes de connaissances

Le graphe de connaissance est une structure dont le rôle principal est d'unifier la connaissance. Le nombre croissant de données disponibles en ligne et l'effort fourni par des groupes comme Wikipédia pour structurer ces données bénéficient au développement des KBs (Gutierrez et Sequeda, 2021). Les récents progrès en matière de représentation de la connaissance permettent de compléter les graphes sémantiques pour développer de nouvelles connaissances. Les KBs sont une source de premier choix pour obtenir une meilleure compréhension du langage naturel (Bast *et al.*, 2016).

1.1 *Principes fondamentaux*

1.1.1 *Triplet RDF*

Le triplet RDF (Ressource Description Framework) est la structure la plus détaillée du graphe de connaissance. Un triplet est composé d'un sujet, d'un prédicat et d'un objet : $(sujet, predicat, objet)$. Le graphe de connaissance ou KB désigne l'ensemble des triplets dans lequel les nœuds (entités) sont liés entre elles par des arcs dirigés (relations). Une même ressource peut être sujet, prédicat ou objet de plusieurs triplets.

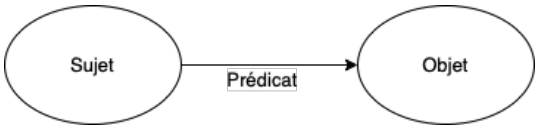


FIGURE 2.1 – Représentation d'un triplet RDF.

1.2 *Graphes de connaissances en tant que ressources*

Freebase, DBpedia [1], Wikidata [2] et Wordnet [3] sont des KBs dont l'ambition est de faciliter l'intégration des connaissances structurées dans des applications. Wordnet est une ressource de référence pour la tâche de compréhension du langage naturel. Freebase, DBpedia et Wikidata sont des encyclopédies universelles structurées.

1. https://www.dbpedia.org
2. https://www.wikidata.org/wiki/Wikidata
3. https://wordnet.princeton.edu

1.2.1 *Wordnet*

Oram (2001) a créé Wordnet, un KB qui centralise des informations des mots de la langue anglaise. L'ambition de Wordnet est de faciliter le traitement du langage naturel en remplaçant les traditionnels dictionnaires par une structure plus adaptée aux machines. Wordnet centralise des informations sur les noms, les verbes, les adjectifs et les adverbes. Les entités sont organisées selon leurs champs lexicaux et connectés via des relations lexicales variées :

La relation de *Synonymie*, traduit la similarité sémantique entre deux mots. *L'antonymie*, indique l'opposition sémantique entre deux mots. *L'hyponymie*, désigne une relation de subordination entre deux termes. La *méronymie*, traduit l'appartenance d'un mot à un groupe plus large. L'hyponymie, n'est pas systématiquement transitive contrairement à la méronymie. La relation de *troponymie*, désigne un verbe qui détaille l'action décrite par un autre verbe. La relation *d'implication*, indique une conséquence logique. Des exemples sur les relations Wordnet sont présentés dans le tableau 2.1.

Relation	**Catégorie**	**Exemple**
synonymie	noms, verbes, adjectifs, adverbes	chat, matou
antonymie	adjectifs, adverbes	doux, agressif
hyponymie	noms	chat domestique, chat
meronymie	noms	chat, félin
troponymie	verbes	guillotiner, décapiter
implication	verbes	divorce marier

TABLEAU 2.1 – Relations sémantiques définies par Wordnet.

1.2.2 *Freebase*

Freebase (Bollacker *et al.*, 2008) était un KB opéré de manière collaborative entre 2007 et 2014. Il centralise des connaissances encyclopédiques dans un format adapté aux machines à la distinction de Wikipédia qui était initialement textuel. Google suspend le service Freebase en 2015 et transfert les 1.9 milliards de triplets de Freebase vers Wikidata (Pellissier Tanon *et al.*, 2016).

1.2.3 *DBpedia*

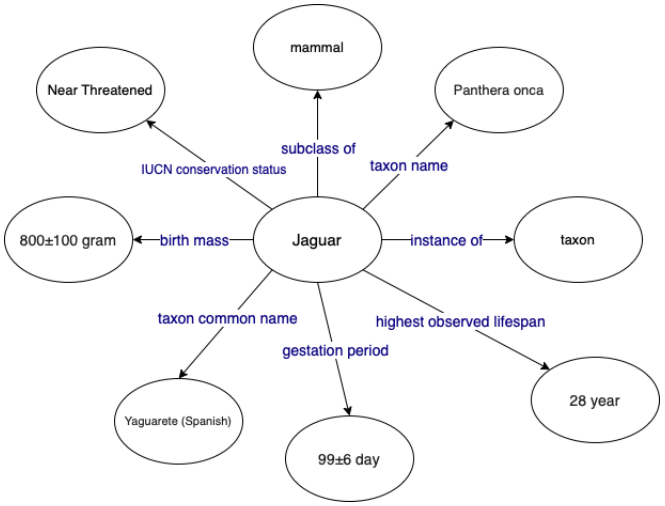
DBpedia (Lehmann *et al.*, 2015) est un KB construit à partir des données structurées de Wikipédia (infoboxes et tableaux de Wikipédia). DBpedia référence plus 228 millions d'entités et 850 millions de faits (triplés) (Holze, 2022). L'objectif de DBpedia est de faciliter l'accès aux données structurées de Wikipédia et d'enrichir les applications des utilisateurs du KB. Le tableau 2.2 présent quelques statistiques sur les types d'entités disponibles sur DBpedia.

Catégorie	Personnes	Lieux	Albums	Films	Jeux Vidéos	Entreprises	Espèces	Plantes	Maladies
Fréquence	1,792,308	748,372	157,566	144,415	24,829	87,621	1,933,436	7718	10,591

TABLEAU 2.2 – Nombre d'entités dans DBpedia en fonction du champs lexical des entités.

1.2.4 *Wikidata*

Wikidata (Vrandečić et Krötzsch, 2014) est un KB qui centralise et structure les données de l'encyclopédie universelle Wikipédia. Wikidata contient plus de 98 millions d'entités. La mise à jour d'une propriété d'une entité de Wikidata corrige le contenu des articles de Wikipédia. En mettant à jour *la population de jaguar dans le monde* au sein de Wikidata, les articles de Wikipédia seront mis à jour dans l'ensemble des langues contribuant ainsi à l'uniformité et à la cohérence de l'encyclopédie. Les connaissances référencées dans Wikidata sont distinctes de DBpedia (Ismayilov *et al.*, 2018). Wikidata est une ressource collaborative qui alimente Wikipédia contrairement à DBpedia qui est construit à partir de Wikipédia. Dans la figure 2.2, il est présenté un exemple d'infobox ainsi que la représentation en forme de graphe des triplets obtenus à partir de cet infobox.



(a) Triplets Wikidata

(b) Infobox Wikipédia

FIGURE 2.2 – Un sous-ensemble de triplets de Wikidata décrivant le jaguar et les méta données associées au Jaguar affichés sur la page Wikipédia du jaguar.

2 Apprentissage des représentations des entités et des relations

Les applications des KBs motivent le développement des méthodes dédiées à l'apprentissage de représentation des connaissances et à la complétion des KBs qui sont incomplets par nature Rossi *et al.* (2021). 71% des personnes décrites dans Freebase n'ont pas de lieu de naissance et 75% d'entre elles n'ont pas de nationalité connue Minervini *et al.* (2015). Une KB peut être étendue par la prédiction des liens inexistants mais cohérents avec sa structure. La tâche de prédiction des liens est un moyen classiquement utilisé pour compléter une KB, et par conséquence, fréquemment utilisée lors de l'évaluation des modèles pour la représentation de KBs. Les modèles de représentation basés sur la prédiction de liens fournissent des représentations sémantiques des entités qui sont utilisées sur une variété de tâches en aval. Dans cette section, nous définirons la procédure d'apprentissage de la tâche de représentation des connaissances. Nous présenterons les caractéristiques des modèles de l'état de l'art et les fonctions d'optimisation associées. Finalement, nous analyserons les stratégies permettant d'apprendre des représentations d'entités et de relations à partir de plusieurs KBs.

2.1 *Prédiction des liens, des relations et des triplets*

Une représentation courante de la connaissance dans les KBs est sous la forme d'un triple, noté (h, r, t), indiquant qu'il existe une relation r entre son entité de tête h et son entité de queue t. Pour capturer la sémantique au sein des KBs, l'idée clé est de représenter les entités et les relations sous forme de vecteurs (appelés embeddings). Un des premiers travaux, TransE (Bordes *et al.*, 2013), interprète une relation comme étant la translation de l'entité principale vers l'entité secondaire. Il s'attend à ce $\mathbf{h} + \mathbf{r} \simeq \mathbf{t}$ si le triplet $(h, r, t) \in \mathcal{GC}$ se vérifie, où $(\mathbf{h}, \mathbf{r}, \mathbf{t}) \in \mathbb{R}^k$ sont les vecteurs de dimension $k \in \mathbb{R}$ qui désignent les entités et la relation d'un triplet. TransE s'est imposé comme une technique fondamentale pour la représentation des KBs.

2.1.1 *Définition de la tâche*

La tâche de prédiction des liens consiste à exploiter les triplets d'un KB pour en déduire de nouveaux. Cela revient à proposer les entités les plus susceptibles de compléter le triplet *(h, r, ?)* ou *(?, r, t)*. $\mathcal{E}$ désigne l'ensemble des entités et $\mathcal{R}$ l'ensemble des relations du KB. $\mathcal{T}^+$ comprend l'ensemble des triplets du KB tel

que $\mathcal{T}^+ \in \mathcal{E} \times \mathcal{R} \times \mathcal{E}$. $\mathcal{T}^-$ est l'ensemble des triplets corrompus qui ne sont pas référencés dans le KB tel que $\mathcal{T}^- \in \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ avec $\mathcal{T}^- \notin \mathcal{T}^+$. Les modèles de prédiction des liens sont entraînés à distinguer des triplets $\in \mathcal{T}^+$ des triplets corrompus $\in \mathcal{T}^-$. Les modèles sont entraînés à produire des scores élevés pour un triplet (h, r, t) $\in \mathcal{T}^+$ et un score inférieur pour un triplet corrompu (h, r, t') $\in \mathcal{T}^-$.

Type	Model	Fonction de score
Modèles géométrique	TransE (Bordes *et al.*, 2013)	$\|\mathbf{h} + \mathbf{r} - \mathbf{t}\|_2^2$
	TransH (Wang *et al.*, 2014b)	$\| \left(\mathbf{h} - \mathbf{w}_r^\top \mathbf{h} \mathbf{w}_r\right) + \mathbf{d}_r - \left(\mathbf{t} - \mathbf{w}_r^\top \mathbf{t} \mathbf{w}_r\right) \|_2^2$
	TransR (Lin *et al.*, 2015a)	$\|\mathbf{M}_r \mathbf{h} + \mathbf{r} - \mathbf{M}_r \mathbf{t}\|_2^2$
	TransD (Ji *et al.*, 2015)	$\|\mathbf{h}_\perp + \mathbf{r} - \mathbf{t}_\perp\|_2^2$
	STransE (Nguyen *et al.*, 2016)	$\|\mathbf{M}_{r,1} \mathbf{h} + \mathbf{r} - \mathbf{M}_{r,2} \mathbf{t}\|_2^2$
	TorusE (Ebisu et Ichise, 2018)	$2d_{L_1}([\boldsymbol{h}] + [\boldsymbol{r}], [\boldsymbol{t}])$
	RotatE (Sun *et al.*, 2019b)	$\|\boldsymbol{h} \odot \boldsymbol{r} - \boldsymbol{t}\|_2^2$
Modèles de décomposition en tenseurs	DistMult (Yang *et al.*, 2015b)	$\langle \boldsymbol{h}, \boldsymbol{r}, \boldsymbol{t} \rangle$
	ComplEx (Trouillon *et al.*, 2016)	$\langle \boldsymbol{r}, \boldsymbol{h}, \overline{\boldsymbol{t}} \rangle$
	Analogy (Liu *et al.*, 2017)	$\langle \mathbf{h}, \mathbf{r}, \mathbf{t} \rangle$
	SimplE (Kazemi et Poole, 2018)	$\frac{1}{2}(\langle \mathbf{h}_h, \mathbf{r}, \mathbf{t}_t \rangle + \langle \mathbf{h}_t, \mathbf{r}^{-1}, \mathbf{t}_h \rangle)$
	HolE (Nickel *et al.*, 2016)	$\sigma\left(\boldsymbol{r}^\top (\boldsymbol{h} \star \boldsymbol{t})\right)$
	TuckER (Balazevic *et al.*, 2019)	$\sigma(\mathcal{W} \times \mathbf{h} \times \mathbf{r} \times \mathbf{t})$
Réseaux de neurones	ConvE (Dettmers *et al.*, 2018)	$g(\boldsymbol{W} \times g([\boldsymbol{h}; \boldsymbol{r}] \circledast \boldsymbol{\omega}) + \mathbf{b}) \times \boldsymbol{t}$
	ConvKB (Nguyen *et al.*, 2018)	$g(\boldsymbol{W} \times g([\boldsymbol{h}; \boldsymbol{r}; \boldsymbol{t}] \circledast \boldsymbol{\omega}) + \mathrm{b})$
	ConvR (Jiang *et al.*, 2019)	$g\left(\boldsymbol{W} \times g\left([\boldsymbol{h}] \circledast \boldsymbol{\omega}_r\right) + \mathrm{b}\right) \times \boldsymbol{t}$
	CapsE (Nguyen *et al.*, 2019)	$\|\operatorname{capsnet}(g([\boldsymbol{h}; \boldsymbol{r}; \boldsymbol{t}] \circledast \boldsymbol{\omega}))\|_2^2$
	RSN (Guo *et al.*, 2019)	$\sigma(r \sin(\boldsymbol{h}, \boldsymbol{r}) \times \boldsymbol{t})$

TABLEAU 2.3 – Taxonomie des fonctions de score des modèles de prédiction de liens basée sur les travaux de Rossi *et al.* (2021).

2.2 *Modèles de prédiction des liens pour la représentation des entités et des relations*

La plupart des modèles dédiés à la représentation des entités et des relations des KBs définissent une fonction de score f afin d'évaluer la vraisemblance d'un triplet (h, r, t). Les modèles construisent des représentations vectorielles des entités et des relations afin d'alimenter la fonction de score f. Les recherches récentes en matière de prédiction de liens peuvent être regroupées dans trois types de modèles (Rossi *et al.*, 2021) : 1) Les *modèles géométriques* qui interprètent les relations comme des opérations géométriques dans l'espace latent. 2) Les modèles de *décomposition en tenseur* interprètent la similarité entre les entités et les relations via un produit de vecteurs. 3) Les *approches basées sur les réseaux de neurones* qui s'appuient sur les récents progrès en apprentissage profond pour représenter les entités et les relations. Les KBs peuvent contenir des millions de faits ; par conséquent, les modèles doivent être capables de gérer la mise à l'échelle tant en ce qui concerne le nombre de paramètres que les coûts de calcul pour être applicables dans des scénarios du monde réel.

2.2.1 *Modèles géométriques*

TransE

TransE (Bordes *et al.*, 2013) est le premier modèle géométrique dédié à la tâche de prédiction des liens. TransE produit des représentations des entités et des relations de sorte que le vecteur de l'objet soit proche de la somme des vecteurs de la relation et du sujet. Pour apprendre une représentation des entités et des relations, TransE minimise une fonction de classement basée sur une marge. Concrètement, le modèle doit produire des scores plus élevés pour un triplet existant $(h, r, t) \in \mathcal{T}^+$ par rapport à un triplet corrompu $(h', r, t') \in \mathcal{T}^-$. L'ensemble de triplets corrompus $\mathcal{T}^-$ est composé de triplets dont la tête ou la queue est remplacée par une entité aléatoire.

$$\mathcal{L}_{TransE} = \sum_{(h,r,t)\in mathcal T^+} \sum_{(h',r',t')\in T^-_{(h,r,t)}} \left[\gamma + f(h,r,t) - f\left(h',r,t'\right)\right] \tag{2.1}$$

γ dénote le paramètre de marge qui permet d'étendre l'espace des représentations des entités et des relations.

Le modèle TransE s'applique bien aux relations 1 à 1 mais a des problèmes pour les relations les relations $N-1$, $1-N$ et $N-N$ (Lin *et al.*, 2015b). Prenons une relation

r de type $N-1$, $\forall i \in \{0,\ldots,m\}, (h_i,r,t) \in \mathcal{T}^+$. Dans le cadre de l'apprentissage de la relation r, l'entraînement de TransE produira des représentations identiques pour l'ensemble des entités $\{h_0,..,h_m\}$.

TransH

Le modèle TransH (Wang *et al.*, 2014b) projette les entités dans un hyperplan spécifique à chaque relation à partir d'une matrice $\mathcal{W}_r$ avant d'exécuter la fonction de score. La fonction objectif de TransH (équation 2.3) étend la fonction objectif de TransE et propose les contraintes de régularisation 1,2,3 :

$$\forall e \in \mathcal{E}, \|\mathbf{e}\|_2 \leq 1 \tag{1}$$

$$\forall r \in \mathcal{R}, \|\mathbf{w}_r\|_2 = 1 \tag{2}$$

$$\forall r \in \mathcal{R}, \left|\mathbf{w}_r^\top \mathbf{d}_r\right| / \|\mathbf{d}_r\|_2 \leq \epsilon \tag{3}$$

La contrainte 1 tend la norme des représentations de chaque entité à être inférieur à 1. La contrainte 2 tend $\mathbf{w}_r$ vers un vecteur unitaire. Finalement, la contrainte 3 garantit que le vecteur de traduction $\mathbf{d}_r \in \mathbb{R}^k$ est dans l'hyperplan $\mathbf{w}_r \in \mathbb{R}^k$. Pour rappel la fonction de score de TransH , définie dans le tableau 2.3, est égale à :

$$f_{TransH}(h,r,t) = \left\| \left(\mathbf{h} - \mathbf{w}_r^\top \mathbf{h} \mathbf{w}_r\right) + \mathbf{d}_r - \left(\mathbf{t} - \mathbf{w}_r^\top \mathbf{t} \mathbf{w}_r\right) \right\|_2^2 \tag{2.2}$$

La fonction objectif de TransH est formalisée par l'équation :

$$\begin{aligned} \mathcal{L}_{TransH} = & \sum_{(h,r,t) \in mathcal T^+} \sum_{(h',r',t') \in T^-_{(h,r,t)}} \left[\gamma + f(h,r,t) - f\left(h',r,t'\right)\right] \\ & + C \left\{ \sum_{e \in \mathcal{E}} \left[\|\mathbf{e}\|_2^2 - 1\right]_+ + \sum_{r \in \mathcal{R}} \left[\frac{\left(\mathbf{w}_r^\top \mathbf{d}_r\right)^2}{\|\mathbf{d}_r\|_2^2} - \epsilon^2 \right]_+ \right\} \end{aligned} \tag{2.3}$$

(h,r,t) fait référence à un triplet existant $\in \mathcal{T}^+$ par rapport à un triplet corrompu $(h',r,t') \in \mathcal{T}^-$. Le paramètre C permet de pondérer l'importance des contraintes 1, 2 et 3 par rapport à l'objectif primaire de modélisation des entités et des relations.

TransR

TransE et TransH représentent tous deux les entités et les relations dans le même espace $\mathbb{R}^k$. Lin *et al.* (2015b) considèrent les relations et les entités comme des objets complètement différents. Ainsi le modèle TransR propose de représenter les entités et les relations dans deux espaces distincts.

La fonction de score de TransR est :

$$f_{TransR}(h, r, t) = \|\mathbf{h}\mathbf{M}_r + \mathbf{r} - \mathbf{t}\mathbf{M}_r\|_2^2 \tag{2.4}$$

Avec $\mathbf{h}, \mathbf{t} \in \mathbb{R}^k, \mathbf{r} \in \mathbb{R}^d$. La matrice $\mathbf{M}_r \in \mathbb{R}^{k \times d}$ projete les entités dans l'espace des relations.

Notez que la fonction objectif du modèle TransR est identique à celle de TransE.

TransD

Le modèle TransD (Ji *et al.*, 2015) propose un opérateur de projection qui dépend à la fois des entités et des relations contrairement à la projection de TransR qui dépend exclusivement des relations. De façon similaire à TransR, les auteurs de TransD choisissent de représenter les entités et les relations dans un espace distinct.

La fonction de score de TransD est formalisée par :

$$f_{TransD}(h, r, t) = \|\mathbf{h}_\perp + \mathbf{r} - \mathbf{t}_\perp\|_2^2 \tag{2.5}$$

Avec $\mathbf{h}_\perp, \mathbf{t}_\perp \in \mathbb{R}^m$, les représentations des entités projetées dans l'espace des relations et définies par :

$$\begin{aligned} \mathbf{h}_\perp &= \mathbf{M}_{rh}\mathbf{h} \\ \mathbf{t}_\perp &= \mathbf{M}_{rt}\mathbf{t} \end{aligned} \tag{2.6}$$

$\mathbf{h}, \mathbf{t} \in \mathbb{R}^k$ désignent les représentations de la tête et de la queue du triplet (h, r, t). $\mathbf{M}_{rh}, \mathbf{M}_{rt} \in \mathbb{R}^{m \times k}$ sont les matrices de projection des entités dans l'espace des relations.

Les matrices de projection des entités sont définies par :

$$\begin{aligned} \mathbf{M}_{rh} &= \mathbf{r}_p \mathbf{h}_p^\top + \mathbf{I} \\ \mathbf{M}_{rt} &= \mathbf{r}_p \mathbf{t}_p^\top + \mathbf{I} \end{aligned} \tag{2.7}$$

$\mathbf{r}_p \in \mathbb{R}^m$ désigne le vecteur de projection associé à la relation r. $\mathbf{h}_p^\top, \mathbf{t}_p^\top \in \mathbb{R}^k$ sont les vecteurs de projection des entités h, et t. La matrice identité $\mathbf{I} \in \mathbb{R}^{m \times k}$ est utilisée pour initialiser les matrices de projections $\mathbf{M}_{rh}$ et $\mathbf{M}_{rh}$.

Le modèle TransD suit la fonction d'optimisation standard de TransE.

STransE

STransE (Nguyen *et al.*, 2016) est une extension du modèle TransR. TransR initialise une matrice $\mathbf{M}_r \in \mathbb{R}^{k \times d}$ dédiée à la projection des entités dans l'espace des relations $\mathbf{r} \in \mathbb{R}^d$. STransE étend le modèle TransR via le remplacement de la matrice $\mathbf{M}_r$ par deux matrices $\mathbf{M}_{r1}, \mathbf{M}_{r2} \in \mathbb{R}^{k \times k}$. La matrice $\mathbf{M}_{r1}$ projette la représentation de la tête du triplet $\mathbf{h} \in \mathbb{R}^k$ et la matrice $\mathbf{M}_{r1}$ projette la queue $\mathbf{t} \in \mathbb{R}^k$ du triplet. Nous pouvons noter que la dimension de l'espace de représentations des entités et des relations est identique dans STransE contrairement à TransR.

La fonction de score de STransE est définie par :

$$f_{STransE}(h, r, t) = \|\mathbf{M}_{r,1}\mathbf{h} + \mathbf{r} - \mathbf{M}_{r,2}\mathbf{t}\|_2^2 \tag{2.8}$$

Le modèle STransE poursuit la fonction objectif défini par TransE dans le cadre de son apprentissage.

TorusE

Ebisu et Ichise (2018) signalent les problèmes de régularisation du modèle TransE. La régularisation est nécessaire à la convergence de TransE et permet de définir des limites à l'espace défini par le modèle mais déforme les représentations des entités et les force à être sur une sphère dans l'espace vectoriel de TransE. Pour remédier à ces difficultés, les auteurs présentent le modèle TorusE. Ils opèrent l'apprentissage dans un espace de type torus $\mathbb{T}^k$ avec k dimensions. Le torus est un espace borné et ne nécessite pas de régularisation pour représenter les entités et les relations.

Les auteurs présentent une mesure de distance d_{torus} adaptée à l'espace du torus qu'ils intègrent dans la fonction de score de TorusE :

$$d_{torus}([\boldsymbol{x}],[\boldsymbol{y}]) = \sum_{i=1}^{n} \min\left(\left|\pi_{frac}\left(x_i\right) - \pi_{frac}\left(y_i\right)\right|, 1 - \left|\pi_{frac}\left(x_i\right) - \pi_{frac}\left(y_i\right)\right|\right) \quad (2.9)$$

Avec $[\boldsymbol{x}],[\boldsymbol{y}] \in \mathbb{T}^k$. π_{frac} est une fonction qui retourne la partie fractionnaire : $\pi_{frac}(x) = x - \lfloor x \rfloor$ et $\lfloor x \rfloor$ désigne le plus grand nombre entier inférieur à x.

La fonction de score de TorusE est définie par :

$$f_{TorusE}(h,r,t) = 2d_{L_1}([\boldsymbol{h}] + [\boldsymbol{r}], [\boldsymbol{t}]) \quad (2.10)$$

Avec les représentations des entités et de la relation $[\boldsymbol{h}],[\boldsymbol{t}],[\boldsymbol{r}] \in \mathbb{T}^n$.

Le modèle TorusE poursuit la fonction objectif définie par TransE dans le cadre de son apprentissage.

RotatE

Sun *et al.* (2019b) proposent le modèle RotatE dont l'objet est de modéliser les relations de symétrie, d'antisymétrie, inverse et de composition. Le modèle RotatE définit chaque relation comme une rotation de l'entité source vers l'entité cible dans l'espace vectoriel.

Une relation r est symétrique si $\forall e_1, e_2$:

$$r(e_1,e_2) \Rightarrow r(e_2,e_1) \quad (2.11)$$

Les relations r_1 et r_2 sont inverse si $\forall e_1, e_2$:

$$r_2(\mathrm{e}_1,\mathrm{e}_2) \Rightarrow r_1(\mathrm{e}_2,\mathrm{e}_1) \quad (2.12)$$

La relation r_1 est composée de la relation r_2 et de la relation r_3 si $\forall e_1, e_2, e_3$.

$$r_2(\mathrm{e}_1,\mathrm{e}_2) \wedge \mathrm{r}_3(\mathrm{e}_2,\mathrm{e}_3) \Rightarrow \mathrm{r}_1(\mathrm{e}_1,\mathrm{e}_3) \quad (2.13)$$

La fonction de score de RotatE est :

$$f_{RotatE}(h,r,t) = \|\mathbf{h} \circ \mathbf{r} - \mathbf{t}\|_2^2 \tag{2.14}$$

Avec les représentations des entités $\mathbf{h}, \mathbf{t} \in \mathbb{C}^k$, la représentation de la relation $\mathbf{r} \in \mathbb{C}^k$ du triplet et $\circ$ correspond au produit de Hadamard.

Pour deux vecteurs de mêmes dimensions $\mathbf{h}, \mathbf{r} \in \mathbb{C}^k$, le produit de Hadamard $\mathbf{h} \circ \mathbf{r}$ produit un vecteur dont les coefficients sont :

$$(\mathbf{h} \circ \mathbf{r})_i = \mathbf{h}_i \times \mathbf{r}_i \tag{2.15}$$

Les auteurs proposent une nouvelle fonction objectif pour leur modèle :

$$\begin{aligned}\mathcal{L}_{RotatE} = \sum_{(h,r,t)\in\mathcal{T}^+} & -\log\sigma\left(\gamma - f(h,r,t)\right) - \\ & \sum_{(h',r',t')\in T^-} \left[p\left(h',r,t'\right)\log\sigma\left(f\left(h',r,t'\right) - \gamma\right)\right]\end{aligned} \tag{2.16}$$

où γ est une marge fixe, σ est la fonction sigmoïde. (h,r,t) désigne un triplet existant de la base de connaissance $\mathcal{T}^+$. (h',r,t') désigne un triplet corrompu $\in T^-$. Cette fonction objectif propose également une nouvelle approche d'échantillonnage pour la création de triplets corrompus en filtrant les triplets négatifs qui n'apportent pas d'information au modèle.

Les auteurs définissent une stratégie pour échantillonner les triplets négatifs à partir d'un triplet existant :

$$p\left(h',r,t' \mid \{(h,r,t)\}\right) = \frac{\exp \alpha f\left(h',r,t'\right)}{\sum_j \exp \alpha f\left(h'_j,r,t'_j\right)} \tag{2.17}$$

Avec le paramètre $\alpha \in \mathbb{R}$ qui correspond à la température de la distribution.

2.2.2 *Décomposition en tenseurs*

Les modèles de décomposition en tenseurs pour la prédiction de liens visent à apprendre des représentations d'entités et de relations dont le produit des représentations fournit un score élevé pour les faits existants. Parmi les plus relevants,

nous pouvons citer DistMult (Yang *et al.*, 2015b), ComplEx (Trouillon *et al.*, 2016), Analogy (Liu *et al.*, 2017), SimplE (Kazemi et Poole, 2018), HolE (Nickel *et al.*, 2016) et TuckER (Balazevic *et al.*, 2019).

2.2.3 *Modèles neuronaux*

Les approches d'apprentissage neurales utilisent un réseau de neurones et permettent d'apprendre des paramètres communs à l'ensemble des entités et des relations. Nous trouvons ici les modèles ConvE (Dettmers *et al.*, 2018), ConvKB (Nguyen *et al.*, 2018), ConvR (Jiang *et al.*, 2019) et CapsE (Nguyen *et al.*, 2019).

2.3 *Apprentissage de représentations multi-KB*

Étant donné une collection de graphes de connaissances, l'apprentissage relationnel multigraphes se réfère à la tâche d'apprentissage de représentations riches en informations des entités et des relations à travers les graphes. L'objectif principal de l'apprentissage de la représentation multigraphes est de permettre aux modèles de créer des liens entre les entités et les relations à partir de différents contextes sémantiques. Les méthodes d'apprentissages multigraphes sont particulièrement adaptées à l'alignement des graphes. En effet, encoder différents graphes dans un espace de représentation unifié permet de mesurer la probabilité d'alignement entre entités. Le raisonnement collaboratif à partir de plusieurs KBs comprend non seulement l'inférence, la validation de connaissances, mais aussi la détection de conflits, c'est-à-dire l'identification de connaissances erronées ou de conflits entre des connaissances (Dong *et al.*, 2015; Zhao *et al.*, 2020). La conception des modèles de représentation multi-KB est généralement dédiée à l'alignement de graphes (Liu *et al.*, 2016; Trivedi *et al.*, 2018; Zhu *et al.*, 2017a; Sun *et al.*, 2018) ou l'apprentissage de représentations multilingues (Chen *et al.*, 2017, 2018; Zhang *et al.*, 2019b).

2.3.1 *LinkNBed*

Trivedi *et al.* (2018) proposent le modèle d'apprentissage de représentation multigraphes LinkNBed basé sur une architecture neuronale profonde. LinkNBed apprend conjointement des représentations des entités de multiples graphes dans un espace partagé. La stratégie LinkNBed repose sur l'intersection des entités des KBs. Ils proposent un objectif multitâche à leur modèle : 1) distinguer des triplets existants de triplets corrompus au sein de chaque KB 2) distinguer des triplets exacts

générés à partir de l'intersection des KBs par rapport à des triplets corrompus générés à partir des KBs.

Le modèle LinkNBed intègre les informations contextuelles comme les caractéristiques des attributs des entités et les informations des types d'entités pour expliquer la relation entre deux entités. Les auteurs utilisent le modèle paragraph2vec (Le et Mikolov, 2014) pour construire les représentations des attributs. Les représentations du sujet et de l'objet d'un triplet sont constituées de l'agrégation de la représentation de l'entité, de ses attributs et du type de l'entité.

L'objectif global du modèle est formulé par la pondération de l'apprentissage indépendant et conjoint des KBs via un coefficient α.

$$\mathcal{L} = \sum^{N} [\alpha \cdot L_{mono} + (1 - \alpha) \cdot L_{multi}] + \lambda \|\theta\|_2^2 \tag{2.18}$$

Avec $\lambda \|\theta\|_2^2$, l'opération de régularisation des poids du modèle.

La procédure d'apprentissage du modèle LinkNBed obtient des résultats supérieurs sur les tâches de prédiction des liens par rapport aux modèles traditionnels qui sont entraînés sur des graphes individuels.

2.3.2 *IPTransE*

Le modèle IPTransE (Zhu *et al.*, 2017b) est dédié à l'alignement des entités via la représentation conjointe des connaissances de KBs. IPTransE encode conjointement les entités et les relations des KBs dans un espace sémantique commun à partir d'un ensemble d'entités alignées. La procédure d'entraînement du modèle IPTransE se décompose en trois modules. Le premier module est un objectif standard visant à apprendre les représentations des entités et des relations dans les KBs. Le second module est un objectif de régularisation qui consiste à faire correspondre les représentations des entités alignées et connues. Le troisième module vise à aligner de façon itérative de nouvelles entités entre les KBs lorsque le modèle est suffisamment confiant.

L'objectif de régularisation est exécuté sur l'ensemble des pairs d'entités alignés entre les KBs :

$$\mathcal{L}_{multi} = \sum_{(e_1, e_2) \in \mathbb{L}} \alpha E\left(e_1, e_2\right) \tag{2.19}$$

Les auteurs considèrent de nouveaux alignements au fur et à mesure de l'entraînement. La procédure d'entraînement aligne les entités e_1 et e_2 si la condition $E(e_1, e_2) < \sigma$ est vérifiée avec σ qui désigne un seuil avec $E(e_1, e_2) = \|\mathbf{e}_1 - \mathbf{e}_2\|_{L1/L2}, \forall e_1 \in KB^1, e_2 \in KB^2$.

Le modèle IPTransE bénéficie de l'apprentissage de plusieurs KBs pour la tâche de prédiction des liens et pour l'alignement des entités.

2.3.3 *EAKG*

Trisedya *et al.* (2019) proposent une procédure dédiée à l'alignement des entités de plusieurs KBs afin d'apprendre des représentations de connaissances multigraphes. Pour ce faire, les auteurs commencent par identifier les relations qui sont partagées par l'ensemble des KBs $\mathcal{R}^{KB^1} \cap \mathcal{R}^{KB^2}$. Si les labels de ces relations diffèrent (*né* et *naissance*) les auteurs choisissent un label unique pour les deux relations (*naître*) et créent le graphe $KB^{1,2}$ qui dispose des nouvelles relations et de l'ensemble des triplets de KB^1 et de KB^2.

L'objectif global du modèle est formalisé par :

$$\mathcal{L} = \mathcal{L}_{standard} + \mathcal{L}_{attribut} + \mathcal{L}_{sim} \tag{2.20}$$

Les auteurs proposent finalement d'aligner les entités dont les représentations sont les plus similaires. Pour ce faire, ils calculent la similarité entre une entité $h_1 \in KB^1$ et l'ensemble des entités $h_2 \in KB^2$.

$$h_{map} = \underset{h_2 \in KB^2}{\operatorname{argmax}} \cos(\mathbf{h}_1, \mathbf{h}_2) \tag{2.21}$$

Les entités dont la similarité est inférieure à un seuil σ sont disqualifiées dans la procédure d'alignement.

La procédure EAKG fournit une solution pour faciliter l'alignement des entités dans un contexte d'apprentissage de représentations des connaissances avec plusieurs KBs.

2.3.4 *MTransE*

MTransE (Chen *et al.*, 2017) propose d'apprendre conjointement des représentations de KB dont les langues diffèrent. Dans la mesure où les bases de connaissances sont construites dans plusieurs langues différentes, la réalisation de l'alignement des connaissances interlinguistiques contribue à la cohérence des KBs et permet de traiter les différentes expressions des relations entre entités dans différentes langues. MTransE représente les entités de chaque langue dans un espace dédié. MTransE fournit des vecteurs pour passer d'une espace à l'autre (d'une langue à l'autre). Le modèle est entraîné sur des graphes partiellement alignés, où une partie des triplets sont alignés avec leurs équivalents multilingues.

Chen *et al.* (2018) améliorent le modèle MTransE en initiant un processus d'alignement des entités entre les ressources. Ils proposent un système de co-entraînement combinant plusieurs modèles multilingues pour apprendre à la fois la structure et les descriptions littérales des entités. À chaque itération lors de l'apprentissage, les modèles de graphe et de texte proposent de nouveaux alignements.

3 Procédures d'évaluation et ressources associées

Pour évaluer les modèles de représentations de la connaissance, il est fréquent d'utiliser des procédures intrinsèques aux graphes de connaissances comme la tâche de prédiction des liens, de prédiction des relations et de classification des triplets. Afin de faciliter la reproduction des résultats et l'apprentissage, les modèles sont entraînés sur des sous-échantillons des bases de connaissances. Ces échantillons dédiés à l'évaluation sont composés de trois sous-ensembles, un ensemble de triplets d'entraînement, un autre de validation et finalement, un dernier utilisé pour le test.

3.1 *Évaluation de la prédiction des liens, des relations et de la classification des triplets*

Les modèles dédiés à la tâche de prédiction de liens peuvent être évalués intrinsèquement via trois procédures distinctes : 1) la prédiction de lien, qui consiste à trouver une entité manquante dans un triplet ; 2) La prédiction de relation, qui consiste à trouver la relation qui lie deux entités ; 3) La classification des triplets, qui consiste à déterminer si un triplet est exact ($\in \mathcal{T}^+$) ou non ($\in \mathcal{T}^-$).

3.1.1 *Évaluation de la prédiction des liens*

Les modèles de prédiction des liens sont évalués sur leur capacité à retrouver l'entité la plus susceptible de compléter un triplet. Les jeux de données standards comme FB15K237 et WN18RR (Toutanova *et al.*, 2015; Dettmers *et al.*, 2018) fournissent un ensemble de triplet d'entraînement $\mathcal{T}^+_{train}$, d'évaluation $\mathcal{T}^+_{valid}$ et de test $\mathcal{T}^+_{test}$. Ces ensembles sont strictement distincts. Les modèles de prédictions des liens sont entraînés avec l'ensemble $\mathcal{T}^+_{train}$. L'ensemble $\mathcal{T}^+_{valid}$ permet de sélectionner les paramètres du modèle. L'ensemble $\mathcal{T}^+_{test}$ est utilisé comme une référence pour comparer la précision des différents modèles.

Les modèles de prédiction de liens sont évalués avec les métriques HITS@K, Rang Moyen (MR) et Rang Moyen Réciproque (MRR). La mesure HITS@K mesure la capacité du modèle à proposer comme candidat le sujet ou l'objet dans le top K parmi l'ensemble des entités $\in \mathcal{E}$ à partir d'un tuple *(sujet, relation)* ou *(relation, objet)*. Les mesures MR et MRR évaluent le rang de l'entité attendue parmi l'ensemble des entités $\in \mathcal{E}$.

On distingue deux versions des métriques lors de l'évaluation de la prédiction des liens : 1) Les métriques filtrées éliminent les entités distinctes de la vérité terrain qui forme un triplet $\in \mathcal{T}^+$ parmi l'ensemble des entités candidates. 2) Les métriques non filtrées considèrent l'ensemble des entités comme des candidats et par conséquent pénalisent le modèle s'il propose un fait exact mais dont l'entité ou la relation n'est pas celle attendue.

Sun *et al.* (2020b) ont diagnostiqué une faiblesse dans les procédures d'évaluations des protocoles d'évaluation utilisés par les auteurs des modèles neuronaux ConvKB (Nguyen *et al.*, 2018) et CapsE (Nguyen *et al.*, 2019). Un aspect essentiel de la méthode d'évaluation est de décider comment de départager les triplets ayant le même score. Lors de l'évaluation des candidats s'il y a plusieurs triplets avec le même score, une sélection aléatoire parmi les candidats est classiquement privilégiée.

3.1.2 *Évaluation de la prédiction des relations*

Nous pouvons aussi évaluer la capacité d'un modèle à retrouver la relation $r \in \mathcal{R}$ qui complète un triplet : $(h, ?, t) \in \mathcal{T}^+$ via la tâche de prédiction de relations (Trouillon *et al.*, 2016). Les métriques utilisées par la tâche de prédiction de relation sont les mêmes que celles utilisées pour la prédiction des liens (HITS@K, MR et MRR).

3.1.3 *Évaluation de la classification des triplets*

Socher *et al.* (2013) ont proposé l'évaluation de la tâche de classification des triplets. L'objectif étant pour le modèle de déterminer si un triplet (h, r, t) $\in \mathcal{T}^+$. La tâche de classification de triplet peut être considérée comme la réponse à la question "Est-ce que le Jaguar est un félin" en évaluant le score proposé par un modèle pour le triplet (jaguar, famille, félin).

Dans le cadre de l'évaluation de la tâche de classification des triplets, Socher *et al.* (2013) proposent une procédure de création de datasets d'évaluation $\mathcal{T}^{cls}_{valid}$, $\mathcal{T}^{cls}_{test}$. Lors de la création de triplets négatifs $\in T^-$, ils limitent l'ensemble des entités candidates à celles qui apparaissent à cette position dans l'ensemble des triplets $\in \mathcal{T}^+$. (jaguar, habitat, forêt de Brocéliande) est un exemple négatif potentiel de (jaguar, habitat, forêt Amazonienne). L'objectif de cette procédure est d'exclure les non-relations évidentes comme (jaguar, habitat, félin) afin de rendre l'évaluation plus difficile. Finalement, le modèle est évalué en fonction du nombre de triplets qu'il classifie correctement.

3.2 *Ressources pour l'évaluation des modèles de représentations des connaissances*

Pour évaluer les modèles dédiés à la représentation de connaissances, les auteurs (Bordes *et al.*, 2013; Dettmers *et al.*, 2018; Toutanova *et al.*, 2015; Wang *et al.*, 2021b) s'appuient sur les bases de connaissances (Wordnet, Freebase et Wikidata) et créent des jeux de données de taille réduite afin de faciliter la reproductibilité des résultats et de faciliter le traitement des données du graphe dans le cadre de travaux scientifiques. Les statistiques des ensembles les plus utilisés sont présentées dans le tableau 2.4.

	Wordnet		**Freebase**		**Wikidata**	
	WN18	**WN18RR**	**FB15k**	**FB15k-237**	**Transductive**	**Inductive**
Entités	40,943	40,943	14,951	14,541	4,594,485	4,579,609
Relations	18	18	1,345	237	822	822
Train	141,442	86,835	483,142	272,115	20,614,279	20,496,514
Validation	5,000	3,034	50,000	17,535	5,163	6,699
Test	5,000	3,134	59,071	20,466	5,133	6,894

TABLEAU 2.4 – Métadonnées des KBs dédiées à l'évaluation des modèles de représentation des connaissances.

3.2.1 *WN18*

Bordes *et al.* (2013) proposent un jeu de données dédié à la construction de représentations sémantiques du vocabulaire anglais à partir de Wordnet appelé WN18 [4]. WN18 est une ressource standard pour l'évaluation de méthodes de construction de représentations sémantiques des entités. Le tableau 2.5 présent les types de relations présentes dans cet ensemble ainsi comme sa fréquence et quelques exemples de triplet.

4. WN18 est disponible en téléchargement à l'adresse https://deepai.org/dataset/wn18.

Relation	% Triplets	Exemple
_hyponym	24.63	[plaything.n.01, _hyponym, swing.n.02]
_hypernym	24.60	[jaguar.n.01, _hypernym, big_cat.n.01]
_derivationally_related_form	21.01	[cover.v.01, _derivationally_related_form, covering.n.02]
_member_meronym	5.23	[primulaceae.n.01, _member_meronym, glaux.n.01]
_member_holonym	5.22	[jaguar.n.01, _member_holonym, panthera.n.01]
_has_part	3.40	[carob.n.02, _has_part, carob.n.01]
_part_of	3.40	[flint.n.03, _part_of, michigan.n.01]
_member_of_domain_topic	2.20	[physics.n.01, _member_of_domain_topic, relativistic.a.01]
_synset_domain_topic_of	2.20	[valence.n.01, _synset_domain_topic_of, biology.n.01]
_instance_hyponym	2.08	[actor.n.01, _instance_hyponym, redford.n.01]
_instance_hypernym	2.07	[kamet.n.01, _instance_hypernym, mountain_peak.n.01]
_also_see	0.92	[write.v.01, _also_see, write_out.v.01]
_verb_group	0.80	[begin.v.06, _verb_group, begin.v.02]
_member_of_domain_region	0.65	[france.n.01, _member_of_domain_region, sextillion.n.01]
_synset_domain_region_of	0.64	[tovarich.n.01, _synset_domain_region_of, soviet_union.n.01]
_synset_domain_usage_of	0.45	[sertraline.n.01, _synset_domain_usage_of, trade_name.n.01]
_member_of_domain_usage	0.44	[united_kingdom.n.01, _member_of_domain_usage, peanut_oil.n.01]
_similar_to	0.06	[aesthetic.s.03, _similar_to, tasteful.a.01]
Total	100	-

TABLEAU 2.5 – Représentation des relations de l'ensemble d'entraînement de WN18 et exemples associés.

3.2.2 *WN18RR*

Dettmers *et al.* (2018) ont diagnostiqué une faiblesse de WN18. Le KB contient des triplets dont les relations sont symétriques et référencées dans la collection d'entraînement et d'évaluation, i.e, (guillotiner, troponyme, décapiter) et (décapiter, troponyme, guillotiner). Ils ont créé la collection WN18RR [5] pour garantir la qualité de l'évaluation des modèles de représentation des connaissances en filtrant les triplets dupliqués.

3.2.3 *FB15k*

À partir d'un sous-ensemble de Freebase, Bordes *et al.* (2013) ont construit la collection FB15k [6] dont l'objectif est d'évaluer des modèles de représentation de la connaissance. FB15k propose 1 345 relations variées qui renseignent par exemple sur le style de musique d'un artiste via la relation "*/music/artist/genre*" ou sur la profession d'une personne "*/people/person/profession*". Le tableau 2.6 référence les 14 sujets les plus représentés du dataset FB15k.

5. WN18RR est disponible en téléchargement à l'adresse https://github.com/villmow/datasets_knowledge_embedding/tree/master/WN18RR/original.

6. FB15k est disponible en téléchargement à l'adresse https://deepai.org/dataset/fb15k.

Sujet	award	film	people	music	location	sports	education	base	olympics	common	tv	government	soccer	organization	Total
% Triplets	24.87	19.86	11.86	7.87	6.07	4.79	4.04	2.58	2.55	2.47	1.93	1.65	1.46	1.43	93.43

TABLEAU 2.6 – 14 sujets les plus représentés dans la collection FB15k.

3.2.4 *FB15k-237*

Le KB FB15k possède des relations redondantes (Toutanova et Chen, 2015) et comporte donc un biais dans l'évaluation des modèles de représentation de la connaissance. Toutanova *et al.* (2015) ont filtré les relations redondantes de FB15k et ont proposé le KB FB15k-237[7] comme un nouveau standard pour l'évaluation des modèles de représentation de connaissances. Le tableau 2.7 présente les 20 relations les plus représentées au sein de FB15k-237. La collection contient un grand nombre d'informations sur les personnalités publiques du cinéma.

Relation	% Triplets
/award/award_nominee/award_nominations./award/award_nomination/award_nominee	5.88
/film/film/release_date_s./film/film_regional_release_date/film_release_region	4.74
/award/award_nominee/award_nominations./award/award_nomination/award	4.47
/people/person/profession	4.02
/film/actor/film./film/performance/film	3.49
/award/award_category/nominees./award/award_nomination/nominated_for	3.48
/award/award_winner/awards_won./award/award_honor/award_winner	3.10
/film/film/genre	2.67
/award/award_nominee/award_nominations./award/award_nomination/nominated_for	2.31
/music/genre/artists	2.16
/award/award_category/winners./award/award_honor/award_winner	2.08
/film/film/other_crew./film/film_crew_gig/film_crew_role	1.95
/location/location/contains	1.91
/people/person/nationality	1.54
/music/performance_role/track_performances./music/track_contribution/role	1.39
/people/person/places_lived./people/place_lived/location	1.37
/people/person/gender	1.37
/common/topic/webpage./common/webpage/category	1.33
/sports/sports_position/players./sports/sports_team_roster/team	1.29
/award/award_winning_work/awards_won./award/award_honor/award	1.22
Total	51.76

TABLEAU 2.7 – 10 relations les plus représentées de FB15k-237 parmis les 237 relations du dataset.

7. FB15k-237 est disponible en téléchargement à l'adresse https://www.microsoft.com/en-us/download/details.aspx?id=52312.

3.2.5 *Wikidata5M*

Wikidata5M[8] (Wang *et al.*, 2021b) est un ensemble de données dédié à l'évaluation des modèles de représentation de la connaissance. Le KB intègre un sous-ensemble des entités, des relations et des triplets de Wikidata. Wikidata5M est proposé suivant deux configurations : *inductive* et *transductive*. La configuration *inductive* consiste à évaluer les modèles capables de construire des représentations sémantiques des entités qui ne sont pas présentes dans la collection d'entraînement. La configuration *transductive* est dédiée aux modèles qui n'ont pas la capacité de traiter de nouvelles entités. Les auteurs de Wikidata5M ont aligné les entités du KB avec leur page Wikipédia respective afin de traiter les entités inconnues dans le cadre de la configuration inductive. Les 20 relations le plus fréquentes sont listées dans le tableau 2.8.

Relation	% Triplets	Exemple
instance of	18.68	[government house, montserrat, instance of, Li...
country	6.69	[castillon-savès, country, iso 3166-1 :fr]
country of citizenship	5.58	[José María Alfaro Zamora, country of citizens...
occupation	5.36	[J. R. Claeys, occupation, Politician]
located in the administrative territorial entity	4.46	[rogozina, kołobrzeg county, located in the ad...
member of sports team	4.46	[Junivan De Melo, member of sports team, fk ga...]
place of birth	4.16	[Edwin Thacker, place of birth, Germiston]
given name	4.08	[Alexius of nicaea, given name, Alexios]
cast member	2.48	[Il gatto a nove code, cast member, corrado olmi]
sport	2.36	[Jiang Pengxiang, sport, futebol]
educated at	2.14	[Richard Barnett, educated at, Theodore Roosev...
shares border with	2.02	[bazoches-les-gallerandes, shares border with,...
located in time zone	1.96	[Helenow, Chelm County, located in time zone, ...
taxon rank	1.82	[scotophilus kuhlii, taxon rank, cohesion spec...
genre	1.80	[Tout, tout de suite, genre, Crime Film]
parent taxon	1.67	[hexacinia, parent taxon, Tephritidae]
place of death	1.36	[santiago alvarez (general), place of death, s...
country of origin	1.27	[serenata tragica - guapparia, country of orig...
languages spoken, written or signed	1.16	[António Gedeão, languages spoken, written or ...
participant of	1.08	[hitoshi sogahata, participant of, 2004 Olympi..
Total	74.58	-

TABLEAU 2.8 – Distribution des 20 relations les plus fréquentes du jeu de données inductif Wikidata5M et exemples associés.

8. Wikidata5M est disponible en téléchargement à l'adresse https://deepgraphlearning.github.io/project/wikidata5m.

4 Conclusion

Le rôle du graphe de connaissances est d'unifier et de structurer la connaissance au sein d'une organisation. Le graphe de connaissance est une structure à l'interaction des données et de la connaissance (Gutierrez et Sequeda, 2021).

Les ressources Wordnet (Oram, 2001), Freebase (Bollacker *et al.*, 2008), DBpedia (Lehmann *et al.*, 2015) et Wikidata (Vrandečić et Krötzsch, 2014) permettent d'accéder à des connaissances qui peuvent être intégrées dans une variété d'algorithmes et d'applications. Le moteur de recherche de Google exploite les annotations de Wikidata pour désambiguïser les entités et pour enrichir les résultats des recherches via l'affichage de fiche d'informations (Singhal, 2012).

Les KBs sont incomplets par nature (Minervini *et al.*, 2015). La tâche de prédiction des liens est dédiée à la prédiction des informations manquantes (liens ou relations) entre les entités dans les KBs. Les recherches récentes travaillent par conséquent à développer des modèles de prédiction de liens.

Dans ce chapitre, nous avons distingué trois familles distinctes de modèles : les modèles géométriques, de décomposition en tenseur et les modèles neuronaux (Rossi *et al.*, 2021). Les modèles géométriques interprètent les relations comme des opérations géométriques dans l'espace latent. Les modèles de décomposition en tenseur apprennent des représentations d'entités et de relations dont le produit fournit un score élevé pour les faits existants. Finalement les approches neurales s'appuient sur les récents progrès en apprentissage profond pour capturer des informations globales du graphe et les interactions entre les entités et les relations.

TransE (Bordes *et al.*, 2013) est un des premiers modèles géométriques dédié à la tâche de prédiction des liens. Limité par sa fonction de score, TransE représente difficilement les relations 1-N, N-1, N-N (Lin *et al.*, 2015b). Les progrès en matière de prédiction des relations ont été obtenus via la création de modèle disposant de paramètres supplémentaires dédiés aux entités et relations et de fonctions de score plus élaborées (Wang *et al.*, 2014b; Lin *et al.*, 2015a; Ji *et al.*, 2015; Nguyen *et al.*, 2016; Zhang *et al.*, 2019c). ComplEx (Trouillon *et al.*, 2016), TorusE (Ebisu et Ichise, 2018), RotatE (Sun *et al.*, 2019b) représentent les entités et les relations du KB dans l'espace des complexes plutôt que des réels permettant ainsi de mieux modéliser les relations asymétriques et de faciliter la convergence des modèles. Des travaux récents adaptent les avancées en matière d'apprentissage profond aux KB via des réseaux de neurone convolutionnel comme ConvE (Dettmers *et al.*, 2018), ConvKB (Nguyen *et al.*, 2018), ConvR (Jiang *et al.*, 2019) et CapsE (Nguyen *et al.*, 2019). Pour capturer plus d'information le modèle RSN (Guo *et al.*, 2019) apprend à partir des chaînes d'entités et de relations avec un réseau de neurone récurrent (RNN).

Nous avons aussi étudié les travaux qui ont développé des modèles de représentation multigraphes dont l'objectif est de permettre aux modèles de créer des liens entre les entités et les relations à partir de différents contextes sémantiques. Le raisonnement collaboratif à partir de plusieurs KBs comprend non seulement l'inférence, la validation et la détection de conflits entre les connaissances (Dong *et al.*, 2015; Zhao *et al.*, 2020). Trivedi *et al.* (2018); Sun *et al.* (2018) ont développé des approches permettant de fusionner des graphes de connaissances à partir d'une intersection. Chen *et al.* (2017, 2018) apprennent conjointement des représentations de KBs dont les langues diffèrent pour contribuer à la cohérence des KBs et de développer de nouvelles connaissances.

Pour quantifier la performance des modèles de représentation des connaissances, les auteurs (Socher *et al.*, 2013) définissent des protocoles d'évaluation et proposent des jeux de données composés d'un sous-ensemble d'entités et de relations provenant de KBs afin de faciliter l'évaluation des modèles (Bordes *et al.*, 2013; Dettmers *et al.*, 2018; Toutanova *et al.*, 2015; Wang *et al.*, 2021b).

Les modèles de prédictions de liens produisent des représentations des entités et des relations qui sont utilisées sur une variété de tâches extrinsèque. Ainsi, dans le cadre de cette thèse, nous étudierons comment ces représentations peuvent être utilisées pour augmenter les modèles de langues.

3

MODÈLES DE LANGAGE POUR LE TRAITEMENT DU LANGAGE NATUREL

Introduction

L'objectif d'un modèle de langue pré-entraîné est d'utiliser des documents textes non structurés afin de construire une représentation sémantique qui encode le plus fidèlement possible les documents utilisés lors de l'apprentissage. Les modèles de langue s'appuient sur le contexte d'un mot pour le modéliser car le contexte fournit une bonne approximation du sens des mots. En effet, les mots sémantiquement similaires ont tendance à avoir des distributions contextuelles similaires (Miller et Charles, 1991).

Les cooccurrences lexicales représentent une base utile pour la construction d'espaces sémantiques (Firth, 1957). Les modèles basés sur les cooccurrences sont souvent désignés par le terme *modèle de sémantique distributionnelle* (Clark, 2012; Lenci, 2018). Les approches statistiques ont une longue histoire dans la représentation du langage naturel (Deerwester *et al.*, 1990; Lund et Burgess, 1996; Schütze, 1998). La performance des représentations des modèles basés sur le comptage des mots est cependant limitée par l'absence de supervision. Ils nécessitent par exemple d'appliquer une transformation via des poids attribués en fonction du niveau d'information contenu dans les documents pour mieux correspondre à une tâche extrinsèque (Baroni *et al.*, 2014).

Bengio *et al.* (2000) formulent la modélisation du langage comme une tâche supervisée, où les poids des vecteurs de mots sont calculés pour maximiser la probabilité des contextes dans lesquels le mot est observé. Mikolov *et al.* (2013a) et Pennington *et al.* (2014) proposent Word2Vec et GloVe respectivement, qui représentent une avancée majeure en matière de modélisation du langage naturel.

Le développement de la puissance de calcul, l'émergence des modèles d'apprentissage profond, le développement de procédures d'apprentissage auto-supervisé

et l'accessibilité des corpus de texte ont fait entrer le traitement du langage naturel dans une nouvelle ère (Peters *et al.*, 2018; Vaswani *et al.*, 2017; Devlin *et al.*, 2019).

Les modèles de langage pré-entraînés (PLM) tels que BERT Devlin *et al.* (2019) ont un coût computationnel élevé. Par conséquent, de nombreuses publications ont adapté la distillation de connaissance (Buciluǎ *et al.*, 2006; Hinton *et al.*, 2015a) aux modèles de langage afin de créer des modèles de langage disposant de moins de paramètres tout en conservant leur pouvoir prédictif (Sun *et al.*, 2019a; Jiao *et al.*, 2020; Gou *et al.*, 2021; Tang *et al.*, 2019b; Sanh *et al.*, 2019; Wang *et al.*, 2020b; Liu *et al.*, 2020; Fu *et al.*, 2021).

Non seulement utilisé pour compresser les modèles, le concept de distillation de connaissances via la procédure de *mutual learning* suscite un intérêt croissant pour fusionner les espaces sémantiques distinct et complémentaire (Wu *et al.*, 2021; Zhao *et al.*, 2021).

Dans le cadre de cette thèse, nous étudierons les caractéristiques et les procédures de pré-entraînement des modèles de langage. Nous analyserons les architectures de l'état de l'art permettant de capturer l'information contenue dans les textes. Finalement nous analyserons les travaux permettant de distiller les connaissances des modèles de langage pour d'accélérer leur inférence et améliorer leur précision.

1 Modélisation du langage naturel

La modélisation du langage naturel permet d'analyser le langage humain (Manning et Schutze, 1999). Les recherches récentes en matière de modélisation du langage s'appuient sur les réseaux de neurones et proposent des procédures de pré-entraînement qui permettent de capturer la sémantique. La dernière génération de modèles est basée sur des architectures profondes du type *Transformers* et permet de traiter les mots dans leur contexte (Radford *et al.*, 2018; Devlin *et al.*, 2019). Ensuite, des techniques pour améliorer les procédures de pré-entraînement ont été proposées (Yang *et al.*, 2019; Clark *et al.*, 2020; Lewis *et al.*, 2020a). En particulière, les auteurs montrent qu'augmenter les paramètres de modèles ainsi comme entraîner les modèles sur de plus grandes collections de textes permet une amélioration significative de la performance (Radford *et al.*, 2019; Liu *et al.*, 2019).

1.1 *Embeddings de mots*

1.1.1 *Word2Vec*

Mikolov *et al.* (2013b) proposent les modèles *Skip-gram* et *CBOW* qui sont dédiés à la construction de représentations des mots. L'algorithme Skip-gram apprend la représentation des mots via la prédiction des mots du contexte à partir d'un mot central. L'algorithme CBOW apprend la représentation des mots via la prédiction d'un mot central à partir du contexte.

Étant donné une séquence de mots $w_1, w_2, .., w_T$, Skip-gram minimise la fonction objectif :

$$\mathcal{L} = -\frac{1}{T}\sum_{t=1}^{T}\sum_{\substack{-m<j<m \\ j\neq 0}}^{j} \log P\left(w_{t+j} \mid w_t\right) \tag{3.1}$$

Avec T la taille de la séquence et m un hyperparamètre associé à la taille de la fenêtre de prédiction du modèle. w_t désigne le mot central sélectionné par l'algorithme et w_{t+j} désigne un mot dans le contexte du mot central. La probabilité d'un mot du contexte sachant le mot central est évaluée par la fonction softmax :

$$P(w_{t+j}|w_t) = \frac{exp(\boldsymbol{v}_t^\top \boldsymbol{v}_{t+j})}{\sum_{k\in\mathcal{V}} exp(\boldsymbol{v}_t^\top \boldsymbol{v}_k)} \tag{3.2}$$

Avec $v_t \in \mathbb{R}^k$ la représentation du mot central et $v_{t+j} \in \mathbb{R}^k$ la représentation du mot du contexte. $\mathcal{V}$ désigne un sous-ensemble du vocabulaire.

1.1.2 *GloVe*

GloVe (Pennington *et al.*, 2014) est un modèle bilinéaire qui propose de construire des représentations du langage via la factorisation de la matrice des cooccurrences de mots. Le modèle s'entraîne uniquement sur les éléments non nuls de la matrice de co-occurrence lui permettant d'exploiter efficacement les informations statistiques d'un corpus. Les auteurs du modèle GloVe développent des représentations de mots de sorte que le produit scalaire des représentations des mots $\boldsymbol{w}_i \in \mathbb{R}^k$ et $\tilde{\boldsymbol{w}}_j \in \mathbb{R}^k$ permette d'approcher la valeur associée à leur co-occurrence :

$$\boldsymbol{w}_i^\top \tilde{\boldsymbol{w}}_j = \log\left(X_{ik}\right) \tag{3.3}$$

Avec X_{ij} la valeur à laquelle les mots i et j co-occurent.

La fonction objectif du modèle Glove est exprimée par :

$$\mathcal{L}_{glove} = \sum_{i,j}^{|V|} f\left(X_{ij}\right)\left(\boldsymbol{w}_i^\top \tilde{\boldsymbol{w}}_j - \log X_{ij}\right)^2 \tag{3.4}$$

ou $|V|$ est la taille du vocabulaire, $X \in \mathbb{R}^{|V|\times|V|}$ désigne la matrice des cooccurrences de mots. $w_i \in \mathbb{R}^k$ désigne la représentation du mot i et $\tilde{w}_j \in \mathbb{R}^k$ la représentation du mot j. La fonction $f\left(X_{ij}\right)$ permet de pondérer les co-occurrences trop fréquentes afin de réduire leurs importances :

$$f(X_{ij}) = \begin{cases} \left(X_{ij}/\sigma\right)^\alpha & \text{si } X_{ij} < \sigma \\ 1 & \text{sinon} \end{cases} \tag{3.5}$$

Avec un seuil σ fixé à 100 par les auteurs et un coefficient α fixé à $\frac{3}{4}$.

1.1.3 *FastText*

Le modèle FastText (Bojanowski *et al.*, 2017) est une extension du modèle Skip-gram (Mikolov *et al.*, 2013b). Contrairement au modèle Skip-gram, Fastext pro-

pose de prendre en compte la structure interne de chaque mot. FastText considère chaque mot w comme un ensemble de n-grams $\mathcal{G}_w \subset \{1, \ldots, G\}$. En traitant les sous-unités de mots, les auteurs créent un modèle capable de générer une représentation pour les mots qui ne sont pas mentionnés dans le corpus d'entraînement.

Dans le cadre d'une décomposition en tri-grams, le mot "*where*" est décomposé en un ensemble de sous-unité de mots <*wh, whe, her, ere, re*>. Les balises "<" et ">" permettent de distinguer les préfixes et les suffixes. Les auteurs considèrent aussi le mot original "<*where*>" qu'ils entourent de balises. Ensemble des tri-grams du mots "*where*" :

$$\mathcal{G}_{where} = \{< wh, whe, her, ere, re >, < where >\} \tag{3.6}$$

La représentation d'un mot w est obtenue via la somme des représentations des n-grams $\mathcal{G}_w$ qui le composent. La fonction de score du modèle FastText qui estime la probabilité $P(w_{t+j}|w_t)$ est exprimée par la fonction :

$$s(w_{t+j}, w_t) = \sum_{g \in \mathcal{G}_{w_{t+j}}} \mathbf{v}_g^\top \mathbf{v}_t \tag{3.7}$$

$\mathbf{v}_g \in \mathbb{R}^k$ désigne la représentation d'une des sous unité du mot w_{t+j}. $\mathbf{v}_t \in \mathbb{R}^k$ désigne la représentation du mot central conformément à l'algorithme Skip-gram.

Fonction objectif du modèle FastText :

$$\mathcal{L}_{FastText} = \sum_{t=1}^{T} \left[\sum_{\substack{-m<j<m \\ j \neq 0}}^{j} \ell\left(s\left(w_{t+j}, w_t\right)\right) + \sum_{\tilde{w} \in \mathcal{N}_{w_t}} \ell\left(-s\left(\tilde{w}, w_t\right)\right) \right] \tag{3.8}$$

Avec le mot w_t qui correspond au mot central et w_{t+j} qui correspond au mot du contexte. La fonction ℓ désigne la fonction logistique $\ell : x \mapsto \log\left(1 + e^{-x}\right)$. L'ensemble $\mathcal{N}_{w_t}$ inclus des exemples négatifs échantillonnés à partir du vocabulaire.

1.2 *Modèles de langage pré-entraînés*

Cette section présente des modèles qui proposent de construire une représentation contextualisée des mots. Contrairement aux approches précédentes, les modèles de langage pré-entraînés attribuent à chaque mot une représentation qui est fonction de l'ensemble de la phrase d'entrée. Les représentations sont profondes, dans le sens où elles sont fonction de toutes les couches internes de ces modèles.

1.2.1 *ElMo*

Étant donné une séquence de mots $\{w_1, w_2, .., w_T\}$, le modèle ElMo (Peters *et al.*, 2018) modélise la probabilité du mot w_k à partir de l'ensemble du contexte $\{w_1, .., w_{t-1}, w_{t+1}, .., w_t\}$ de façon bi-directionnelle et minimise la fonction objectif :

$$\mathcal{L}_{ElMo} = -\prod_{t=1}^{T}[\log p\left(w_t \mid w_1, \ldots, w_{t-1}\right) + \log p\left(w_t \mid w_{t+1}, \ldots, w_T\right)] \tag{3.9}$$

Via une architecture de type Bi-LSTM (Graves *et al.*, 2013), ELMo produit $2 \times L + 1$ représentations pour chaque mot de la séquence :

$$\mathcal{H}_m = \{\mathbf{x}_t, \overrightarrow{\mathbf{h}}_{t,j}, \overleftarrow{\mathbf{h}}_{t,j} \mid j = 1, \ldots, L\} = \left\{\mathbf{h}_{t,j} \mid j = 0, \ldots, L\right\} \tag{3.10}$$

$\mathcal{H}_m$ désigne l'ensemble des représentations produites par le modèle. Le scalaire L correspond au nombre de couches associées au Bi-LSTM. $\mathbf{x}_t \in \mathbb{R}^k$ désigne une représentation indépendante du contexte. $\overrightarrow{\mathbf{h}}_{t,j} \in \mathbb{R}^k$ désigne la représentation associée à la partie gauche du contexte et $\overleftarrow{\mathbf{h}}_{t,j} \in \mathbb{R}^k$ la représentation associée à la partie droite du contexte. $\mathbf{h}_{t,j}$ désigne la concaténation de la représentation bi-directionnelle par couche et par mot.

Pour obtenir une représentation unique d'un mot de la séquence, les auteurs calculent une moyenne pondérée des représentations $\mathbf{h}_{t,j} \in \mathbb{R}^k$:

$$\mathbf{h}_m = \gamma \sum_{j=0}^{L} s_j \times \mathbf{h}_{t,j} \tag{3.11}$$

Le vecteur $\mathbf{s} \in \mathbb{R}^L$ contient les poids dédiés à pondérer les représentations $\mathbf{h}_{t,j}$. Les auteurs présentent le scalaire γ comme un levier pour faciliter la convergence du modèle.

Les auteurs montrent que les couches les plus profondes du modèle capturent les informations sur la signification des mots dans leur contexte et peuvent être utilisées pour obtenir de bons résultats dans les tâches désambiguïsation du sens des mots. Les couches proches de l'entrée du modèle capturent des informations relatives à la syntaxe et peuvent être utilisées pour l'étiquetage morphosyntaxique.

1.2.2 *Self-attention et Transformers*

Vaswani *et al.* (2017) présentent le concept de self-attention. La fonction de self-attention est une opération de séquence à séquence : une séquence de vecteurs entre, et une séquence de vecteurs sort. Le rôle du mécanisme de self-attention est de propager les informations associées aux interactions entre les vecteurs d'entrée.

Trois matrices $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V} \in \mathbb{R}^{k \times k}$ interviennent dans le calcul du mécanisme de self-attention. Ces matrices donnent à la couche d'attention des paramètres d'apprentissage qui permettent de modifier les vecteurs entrants $\boldsymbol{x}_i$ en fonction du rôle qu'ils doivent jouer :

$$\mathbf{q}_i = \boldsymbol{Q}\boldsymbol{x}_i \quad \boldsymbol{k}_i = \boldsymbol{K}\boldsymbol{x}_i \quad \boldsymbol{v}_i = \boldsymbol{V}\boldsymbol{x}_i \tag{3.12}$$

Pour faciliter la convergence du modèle et réduire l'instabilité numérique associée au calcul du gradient de la fonction softmax, les auteurs réduisent le produit scalaire $\mathbf{q}_i^\top \mathbf{k}_j$ avec la constante $\sqrt{k}$ qui correspond à la dimension des représentations :

$$\begin{aligned} w'_{ij} &= \frac{\mathbf{q}_i^\top \mathbf{k}_j}{\sqrt{k}} \\ w_{ij} &= \frac{\exp w'_{ij}}{\sum_j \exp w'_{ij}} \\ \boldsymbol{y}_i &= \sum_j w_{ij} \boldsymbol{v}_j \end{aligned} \tag{3.13}$$

La figure 3.1 illustre l'architecture d'une couche de type Transformer. Vaswani *et al.* (2017) multiplient l'exécution parallèle de couches de self-attention et créent

le composant *multi-head attention*. Les couches de self-attention alimentent une couche de normalisation (Ba *et al.*, 2016) puis une série de percéptron multi-couche et finalement une dernière couche de normalisation.

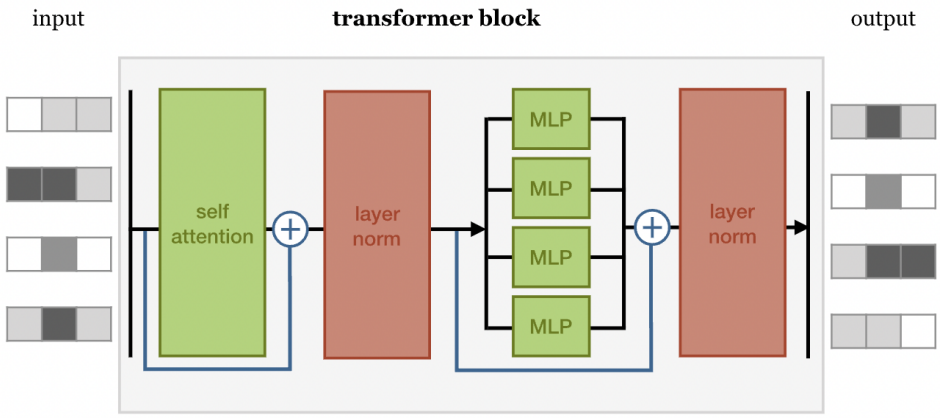


FIGURE 3.1 – Illustration de l'architecture d'un bloc de Transformer (Bloem, 2019).

1.2.3 *BERT*

Le modèle BERT (Devlin *et al.*, 2019) repose sur l'architecture Transformer et il permet de construire une représentation contextualisée des mots. Dans sa version de base, BERT dispose de 12 couches de type Transformer disposant chacune de 12 têtes d'attentions et d'un vocabulaire de taille 30,522, donc soit 110 millions de paramètres au total. Pour alimenter un Transformer, le modèle BERT encode la phrase en entrée via trois couches d'embeddings comme illustrée sur la figure 3.2. Le modèle BERT encode le mot d'une phrase via la somme de la représentation du mot, de la représentation de la phrase et de la représentation de la position du mot dans la phrase.

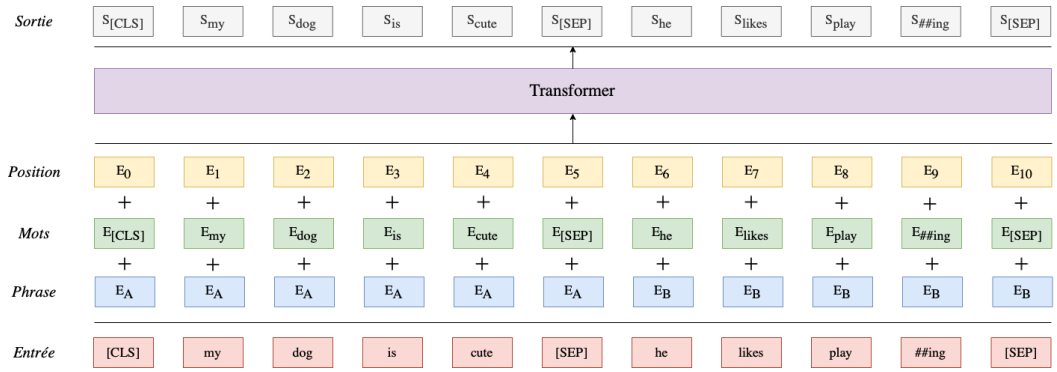


FIGURE 3.2 – Architecture du modèle BERT illustrée.

BERT pré-traite le texte avec un algorithme de découpage des mots lui permettant de conserver un vocabulaire réduit et de traiter les mots qui ne sont pas initialement définis dans son vocabulaire. Comme illustré dans la figure 3.2, les phrases *"My dog is cute. He likes playing."* sont découpés en un ensemble de mots par le modèle : {*[CLS], my, dog, is cute, [SEP], he, likes, play, ##ing, [SEP]*}. La balise "*[CLS]*" permet au transformer de produire une représentation $S_{[CLS]}$ dédiée à la tâche de classification. La balise *[SEP]* est un marqueur pour la fin des phrases. Pour traiter le mot *"playing"* qui n'est pas défini dans le vocabulaire de BERT, le modèle découpe le mot en deux sous-unités de mots : *"play"* et *"##ing"*.

Le modèle BERT est pré-entraîné via les tâches de modélisation du langage masqué (MLM) et de prédiction de phrase consécutive (NSP). Les procédures de MLM et de NSP permettent à BERT d'exploiter l'ensemble du contexte pour réaliser une prédiction ce qui le rend profondément bidirectionnel contrairement à ElMo.

La tâche de MLM consiste à masquer aléatoirement certains des mots de l'entrée et d'entraîner le modèle à retrouver le mot masqué en se basant sur son contexte. Dans le cadre de la procédure de MLM, BERT sélectionne 15% des mots d'une séquence comme les cibles à retrouver. Voici le détail de la procédure de MLM pour la phrase *"Les jaguars mangent des fourmiliers"* lorsqu'on choisit le mot *fourmiliers* comme cible :

— 80% du temps, on remplace le mot *fourmiliers* par [MASK] : *Les jaguars mangent des [MASK].*

— 10% du temps, on remplace le mot *fourmiliers* un mot choisi aléatoirement : *Les jaguars mangent des thèses.*

— 10% du temps, on conserve le mot inchangé : *Les jaguars mangent des fourmiliers.*

La procédure de MLM ne remplace pas systématiquement le mot cible par un masque pour permettre au modèle de généraliser correctement sûr les tâches extrinsèques.

En parallèle de l'objectif de MLM, la tâche de NSP est un objectif binaire qui consiste à prédire si deux phrases A et B sont consécutives. Dans 50% des cas, B est la phrase qui suit réellement A (étiquetée "IsNext"), et dans 50% des cas, il s'agit d'une phrase aléatoire du corpus (étiquetée "NotNext"). L'ambition de la tâche de NSP est d'entraîner le modèle à produire une représentation interne du paragraphe en entré du modèle.

1.2.4 *Évolutions des modèles de langage pré-entraînés (PLM)*

Modèle	Architecture	Pré-entraînement	Paramètres
ELMo	LSTM	BiLM	-
GPT	Transformer Dec.	LM	117M
GPT-2	Transformer Dec.	LM	117M–1.5B
GPT-3	Transformer Dec.	LM	125M–175B
BERT	Transformer Enc.	MLM & NSP	110M–340M
RoBERTa	Transformer Enc.	MLM	355M
XLNet	Two-Stream	PLM	110M–340M
ELECTRA	Transformer Enc.	RTD+MLM	335M
BART	Transformer	DAE	120M–370M

TABLEAU 3.1 – Architectures, procédures de pré-entraînement et nombre de paramètres d'un sous-ensemble des modèles de langues existants.

Les modèles GPT (Radford *et al.*, 2018), GPT-2 (Radford *et al.*, 2019), GPT-3 (Brown *et al.*, 2020) sont des modèles de langage causal basé sur l'architecture Transformer, ils modélisent la probabilité $P(w_t|w_0, w_1, .., w_{t-1})$ dans le cadre de leur pré-entraînement. Le tableau 3.1 présente le nombre de paramètres au sein des différentes versions du modèle GPT. Les auteurs proposent quatre versions du modèle GPT-2 dont le plus grand possède 1.5 milliards de paramètres. Dans sa version la plus large, le modèle GPT-3 dispose de 175 milliards de paramètres. Les auteurs montrent que l'entraînement sur un plus grand ensemble de données et l'utilisation d'un plus grand nombre de paramètres améliorent la capacité du modèle sur une variété de tâches extrinsèques.

Le modèle RoBERTa (Liu *et al.*, 2019) résulte d'une étude approfondie des conditions d'entraînement et des paramètres de BERT. Comparé à BERT, RoBERTa est entraîné plus longtemps, son gradient lors du pré-entraînement est calculé sur plus d'exemples. Les auteurs suppriment l'objectif de prédiction de phrases consécutive et entraînent le modèle sur des phrases plus longues. En outre, RoBERTa dispose de 355 millions de paramètres (voir tableau 3.1).

Le modèle XLNet (Yang *et al.*, 2019) propose une procédure de pré-entraînement appelée procédure de permutation du langage. La tâche de permutation du langage consiste à pré-entraîner un modèle autorégressif et bidirectionnel sur l'ensemble des permutations de mots dans une phrase. Cette procédure conserve les avantages des modèles autorégressifs comme GPT mais permet aussi au modèle

d'être profondément bidirectionnel comme BERT. En outre, les auteurs affirment que le pré-entraînement des modèles autorégressifs est plus similaire aux tâches extrinsèques propres au traitement du langage naturel.

Les auteurs du modèle ELECTRA (Clark *et al.*, 2020) proposent une nouvelle procédure de pré-entraînement appelé détection des mots remplacés (RTD). Plutôt que de masquer aléatoirement des mots dans les phrases, la procédure de RTD consiste à remplacer des mots par des candidats vraisemblables qui sont générés par un modèle entraîné sur la tâche de MLM. Le modèle ELECTRA agit comme un discriminateur dont l'objectif est de prédire si le mot d'une séquence a été remplacé. Les auteurs montrent que cette nouvelle procédure est plus efficace que le MLM car la tâche est définie sur tous les mots qui alimentent le modèle plutôt que sur un sous-ensemble qui a été masqué.

BART (Lewis *et al.*, 2020a) est un modèle génératif de type Transformer et pré-entraîné via une procédure de dé bruitage. Les auteurs corrompent l'ensemble des documents du corpus puis entraînent le modèle à reconstruire les documents originaux. La procédure de bruitage des documents entraîne le modèle à traiter les mots manquants, les mots masqués, les phrases permutées, les morceaux de phrases manquants et les morceaux de phrases masqués. Le modèle obtient des résultats semblables à RoBERTa sur les tâches de classifications et obtient l'état de l'art sur les tâches génératives.

2 Distillation des connaissances

La distillation des connaissances est une procédure dédiée à la compression des modèles (Buciluǎ *et al.*, 2006). Hinton *et al.* (2015a) formulent la distillation comme une fonction objectif qui entraîne un modèle plus léger (étudiant) à reproduire les probabilités fournies par un modèle plus lourd (Professeur). La distillation des connaissances a permis de créer des modèles de langages léger et performants (Tang *et al.*, 2019b; Sanh *et al.*, 2019; Kim et Rush, 2016). Plutôt que de distiller les probabilités en sorties des modèles, Romero *et al.* (2015) adaptent la procédure pour distiller les représentations de réseaux de neurones profond. La distillation des représentations permet de guider le processus d'apprentissage et a été appliquée à la modélisation du langage naturel (Sun *et al.*, 2019a; Jiao *et al.*, 2020; Wang *et al.*, 2020b; Clark *et al.*, 2019). Les recherches récentes en matière de distillation ont mené à la création de nouveaux paradigmes (Zhang *et al.*, 2018, 2019a). La procédure d'apprentissage mutuel (Zhang *et al.*, 2018) met en scène un jeu de rôle dans lesquels un ensemble de modèles joue à la fois le rôle d'étudiant et de professeur. Cette procédure permet de fusionner les connaissances de modèles qui représentent le langage dans des espaces distincts (Wu *et al.*, 2021; Zhao *et al.*, 2021). Zhang *et al.* (2019a) définissent la procédure d'autodistillation qui permet de créer un modèle exécutable à différentes profondeurs permettant ainsi de réaliser un compromis entre vitesse de convergence et performance.

2.1 *Principe de distillation*

Buciluǎ *et al.* (2006) formulent le concept de distillation de connaissances comme une solution pour compresser la connaissance d'un ensemble de modèles (professeurs) dans un unique modèle (étudiant). Les ensembles de modèles permettent de capturer plus d'information (Breiman, 1996; Domingos, 2000; Wolpert, 1992) mais peuvent être trop lents pour répondre aux exigences d'une application. L'objectif de la compression par la distillation de connaissance est donc d'obtenir un unique modèle qui tend vers la performance d'un ensemble de modèles. Buciluǎ *et al.* (2006) utilisent l'ensemble de modèles pour augmenter l'ensemble des données d'entraînements via des pseudo-labels qui d'améliorer l'étudiant.

Hinton *et al.* (2015a) formulent la distillation de connaissances comme une fonction objective. Le rôle de l'étudiant est de reproduire les prédictions des professeurs sous la forme de distributions de probabilités via une mesure de distance ou de divergence. La divergence de Kullback-Leibler est couramment utilisée pour mesurer la divergence entre les prédictions du professeur et de l'étudiant (Gou *et al.*, 2021).

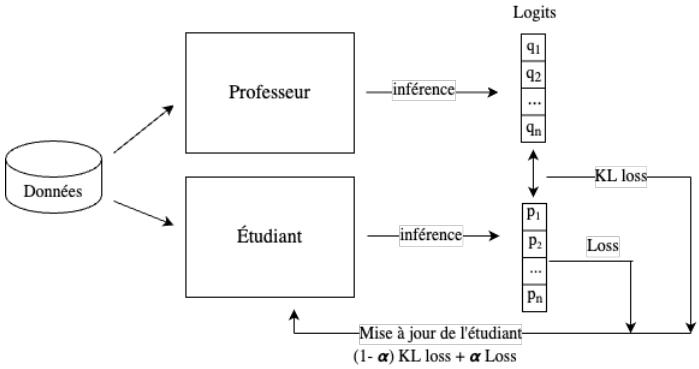


FIGURE 3.3 – Procédure de distillation des connaissances illustrée.

Le schéma 3.3 illustre la procédure de distillation des connaissances définie par Hinton *et al.* (2015a). L'étudiant reproduit les résultats du professeur en minimisant la divergence de Kullback-Leibler avec un coefficient $(1 - \alpha)$ et apprend de lui-même via un coefficient α.

Divergence de Kullback-Leibler (KL) entre les distributions de probabilités $\boldsymbol{q}$ et $\boldsymbol{p} \in \mathbb{R}^k$:

$$KL(\boldsymbol{q}, \boldsymbol{p}) = \sum_i q_i \log \frac{q_i}{p_i} \tag{3.14}$$

Avec q_i et p_i qui dénotent la probabilité d'appartenance à la classe i calculée via une fonction softmax :

$$q_i = \frac{\exp\left(z_i / T\right)}{\sum_j \exp\left(z_j / T\right)} \tag{3.15}$$

La constante T désigne la température de la distillation. L'utilisation d'une valeur plus élevée pour T produit une distribution de probabilité plus lisse.

2.1.1 *Distillation des prédictions pour la tâche intrinsèque de modélisation du langage masqué*

Le Transformer DistilBert (Sanh *et al.*, 2019) présente une procédure de pré-entraînement qui s'appuie sur la distillation des connaissances du modèle Bert.

La fonction objectif de DistilBert est triple, elle combine l'objectif associé à la tâche intrinsèque de MLM (Devlin *et al.*, 2019), l'objectif de distillation $\mathcal{L}_{dis}$ et l'objectif d'alignement des directions des vecteurs de représentation de l'étudiant et du professeur $\mathcal{L}_{cos}$.

$$\begin{aligned} \mathcal{L}_{dis} &= -\sum_i q_i * \log(p_i) \\ \mathcal{L}_{cos} &= -\cos(\boldsymbol{p}, \boldsymbol{q}) \end{aligned} \tag{3.16}$$

Avec les probabilités q_i et p_i associées au mot w_i et estimée par le professeur et l'étudiant via la fonction softmax. La fonction cos désigne la similarité cosinus.

2.1.2 *Distillation des prédictions dédiée à la tâche extrinsèque de classification*

Sur le modèle de la distillation proposée par Hinton *et al.* (2015a), Tang *et al.* (2019b) distillent le modèle BERT dans un modèle de type BiLSTM composé d'une seule couche. Le modèle BiLSTM bénéficie de la procédure distillation sur la tâche extrinsèque. Dans le cadre de la procédure de distillation, les auteurs proposent de minimiser l'erreur quadratique moyenne. La fonction objectif de l'étudiant pour la tâche de classification des sentiments est exprimée par :

$$\mathcal{L}_{dis} = -\alpha \sum_i l_i \log \hat{y}_i - (1-\alpha)\|\boldsymbol{q} - \boldsymbol{p}\|_2^2 \tag{3.17}$$

Avec le paramètre α qui pondère l'apprentissage via la vérité terrain et via la distillation. l_i désigne le label associé à la vérité terrain et $\hat{y}_i$ la probabilité associée estimée par l'étudiant. $\boldsymbol{q}, \boldsymbol{p} \in \mathbb{R}^k$ désignent les distributions de probabilités produites par le professeur et l'étudiant.

Dans le cadre de l'entraînement de l'étudiant, les auteurs créent une procédure d'augmentation des données car un petit ensemble de données peut ne pas suffire pour que l'enseignant transmette pleinement ses connaissances (Ba et Caruana, 2014).

La procédure d'augmentation des données définie par les auteurs pour la distillation de BERT vers un modèle BiLSTM dans le cadre de la tâche de classification des sentiments suit les règles :

— Remplacer avec une probabilité p_{mask} le mot d'une phrase par la balise "[MASK]". "*I loved the comedy*" devient "*I [MASK] the comedy*".
— Remplacer avec une probabilité p_{pos} le mot d'une phrase par un mot dont l'étiquetage morphosyntaxique est identique. "*I loved the comedy*" devient "*I loved the movie*".
— Avec une probabilité p_{ng}, les auteurs sélectionnent un sous-ensemble de l'exemple dont la longueur est variable. "*I loved the comedy*" devient "*I loved*".

Les probabilités p_{mask}, p_{pos}, p_{ng} sont échantillonnées à partir d'une distribution $Uniform(0, 1)$ pour chaque mot w_i d'une phrase $\{w_0, w_1, .., w_T\}$.

2.1.3 *Distillation des prédictions pour la génération de texte*

(Kim et Rush, 2016) étudient la distillation des connaissances pour la tâche de traduction automatique. Les auteurs montrent que la distillation standard des connaissances appliquée à la prédiction au niveau des mots est efficace pour la tâche de traduction automatique. Les auteurs proposent deux nouvelles méthodes permettant de distiller la connaissance à l'échelle d'une séquence de mots. Les auteurs affirment que la distillation des séquences de mots donne à l'enseignant la possibilité de transférer un plus large éventail de connaissances. Leur meilleur étudiant réalise une traduction 10 fois plus rapide que son professeur avec une faible perte de performance et est nettement meilleur qu'un un modèle de taille identique entraîné sans la procédure de distillation des connaissances.

2.2 *Distillation des représentations*

Les réseaux neuronaux profonds sont adaptés à l'apprentissage des représentations (Bengio *et al.*, 2013). Romero *et al.* (2015) s'appuient sur ce constat pour distiller les représentations intermédiaires apprises par le modèle qui joue le rôle de professeur. La distillation des représentations interne guide le processus d'apprentissage du modèle qui joue le rôle de l'étudiant. Les auteurs projettent les représentations de l'étudiant via une matrice W_r pour aligner les représentations du professeur et de l'étudiant lorsque leurs structures diffèrent.

2.2.1 *Patient Knowledge Distillation (PKD)*

Sun *et al.* (2019a) appliquent le concept de distillation des représentations au traitement du langage naturel. Ils proposent une nouvelle fonction objectif basée sur la MSE (erreur quadratique moyenne) entre les représentations des couches de l'étudiant et du professeur. En outre, ils proposent deux stratégies de distilla-

tion. La première stratégie (PKD-Last) consiste à opérer la distillation via les m dernières couches et la seconde (PKD-Skip) consiste à opérer la distillation chaque m couches.

Le modèle PKD combine trois objectifs. Le premier est un objectif standard de classification $\mathcal{L}_{std}$ via la mesure de cross-entropy. Le second objectif $\mathcal{L}_{dis}$ consiste à distiller les prédictions du professeur à l'étudiant via la divergence de Kullback-Leibler. Le troisième objectif $\mathcal{L}_{rpr}$ maximise la similarité entre les représentations du professeur et de l'étudiant via une distance euclidienne.

$$\mathcal{L}_{std} = -\sum_{c \in C} l_i \log \hat{y}_i \tag{3.18}$$

Avec le label l_i qui est égale à 1 si la classe c est vérifiée pour l'exemple i et 0 sinon. La probabilité $\hat{y}_i$ d'appartenance à la classe c est estimée par l'étudiant.

$$\mathcal{L}_{dis} = -\sum_{c \in C} \hat{y}_i^t \log \hat{y}_i \tag{3.19}$$

La probabilité $\hat{y}_i^t$ d'appartenance à classe c est estimée par le professeur.

$$\mathcal{L}_{rpr} = -\sum_{i=1}^{M} \| \frac{\mathbf{h}_i^s}{\|\mathbf{h}_i^s\|_2} - \frac{\mathbf{h}_i^t}{\|\mathbf{h}_i^t\|_2} \|_2^2 \tag{3.20}$$

Avec M qui désigne le nombre de couches du modèle étudiant. $\mathbf{h_j^s}$ désigne la représentation en sortie de la couche j de l'étudiant. $\mathbf{h_t^s}$ désigne la représentation en sortie de la couche j du professeur.

L'objectif global du modèle PKD est une combinaison linéaire composée d'un objectif de classification, un autre de distillation et finalement, une similarité des représentations de l'étudiant et du professeur :

$$L_{PKD} = (1-\alpha)\mathcal{L}_{std} + \alpha\mathcal{L}_{dis} + \beta\mathcal{L}_{rpr} \tag{3.21}$$

2.2.2 *TinyBERT*

La procédure de distillation de TinyBERT (Jiao *et al.*, 2020) est composée d'un objectif triple d'alignement entre : 1) les embeddings de l'étudiant et du profes-

seur ; 2) les représentations des transformers et les matrices d'attentions ; 3) les prédictions de l'étudiant et du professeur. Le modèle TinyBERT réalise la distillation à la fois lors du pré-entraînement et lors de la spécialisation de l'étudiant sur les tâches extrinsèques.

Fonction objectif du modèle TinyBERT :

$$\mathcal{L}_{\text{TinyBERT}} = \sum_{i=1}^{M} \lambda_i \mathcal{L}_{\text{layer}} \left(h_i^s, h_i^t\right)) \tag{3.22}$$

Avec le nombre de couches de l'étudiant M et le scalaire λ_i qui pondère l'objectif en fonction des couches de l'étudiant. La $\mathcal{L}_{\text{layer}}$ qui désigne un objectif distinct qui dépend de la couche de l'étudiant h_i^s :

$$\mathcal{L}_{\text{layer}} = \begin{cases} \mathcal{L}_{\text{embedding}} & m = 0 \\ \mathcal{L}_{\text{attention}} & M \geq m > 0 \\ \mathcal{L}_{\text{prediction}} & m = M + 1 \end{cases} \tag{3.23}$$

L'objectif $\mathcal{L}_{\text{embedding}}$ correspond à l'alignement des embeddings du professeur et de l'étudiant :

$$\mathcal{L}_{\text{embedding}} = -\|\boldsymbol{E}^t - \boldsymbol{E}^s \boldsymbol{W}^E\|_2^2 \tag{3.24}$$

L'objectif $\mathcal{L}_{\text{attention}}$ correspond à une procédure de distillation de l'attention et des représentations des transformers. Le modèle maximise la similarité entre les couches d'attention du professeur et de l'étudiant. De façon complémentaire, le modèle maximise la similarité entre les représentations en sortie des couches de type transformers de l'étudiant et du professeur.

$$\mathcal{L}_{\text{attention}} = -\frac{1}{M_A} \sum_{i=1}^{M_A} [\ \|\boldsymbol{h}_i^t - \boldsymbol{h}_i^s \boldsymbol{W}_i^h\|_2^2 + \frac{1}{|A_i|} \sum_{j=1}^{|A_i|} \|\boldsymbol{A}_{ij}^t - \boldsymbol{A}_{ij}^s\|_2^2\ ] \tag{3.25}$$

Avec M_A qui dénombre les couches de Transformer de l'étudiant. $\boldsymbol{h}_i^t$ désigne la représentation en sortie du transformer associée à la i-ème couche du professeur. La

matrice W_i^h projette les représentations de l'étudiant dans l'espace des représentations du professeur. $|A_i|$ correspond au nombre de têtes d'attention de la couche i. A_{ij}^t désigne la j-ème matrice d'attention de la i-ème couche du professeur et A_{ij}^s de l'étudiant :

$$A_{ij} = \frac{Q_{ij}K_{ij}^T}{\sqrt{d_k}}, \tag{3.26}$$

L'objectif $\mathcal{L}_{\text{prediction}}$ maximise la similarité entre les logits du professeur et de l'étudiant :

$$\mathcal{L}_{\text{prediction}} = -KL(z^t, z^s) \tag{3.27}$$

Avec KL la mesure de divergence de Kullback-Leibler. z^t désigne la distribution de probabilités en sortie du professeur et z^s en sortie de l'étudiant.

2.2.3 *Self-distillation*

Zhang *et al.* (2019a) proposent un nouveau cadre de distillation des connaissances appelé *self-distillation*. Dans le cadre de la procédure de self-distillation (illustrée par le schéma 3.4), un seul et même modèle joue le rôle de professeur et d'étudiant. La self-distillation fournit un modèle capable de réaliser la tâche pour laquelle il a été entraîné à différentes profondeurs. Pour mettre en place la self-distillation, les auteurs décomposent le modèle ResNet50 (He *et al.*, 2016) en 4 blocs et attribuent un classifieur respectif à chacun des blocs. Chaque bloc du ResNet50 dispose d'un triple objectif et est responsable de distiller ses connaissances au bloc qui le précède.

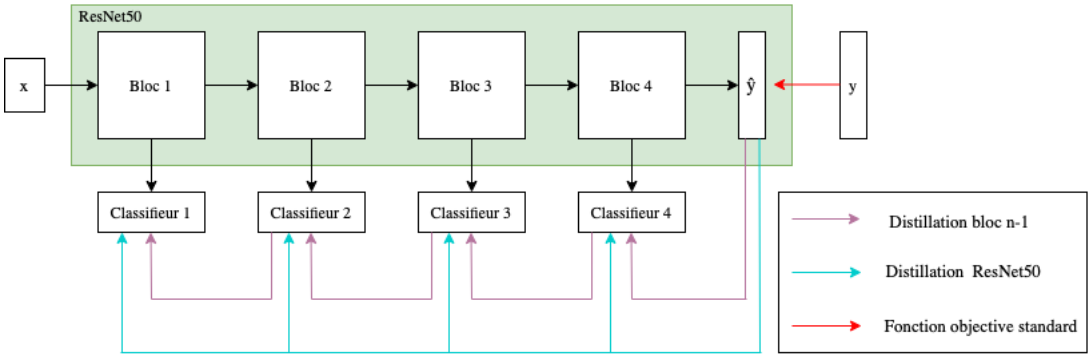


FIGURE 3.4 – Procédure d'autodistillation illustrée.

2.3 *Apprentissage mutuel*

L'apprentissage mutuel est un processus d'apprentissage collaboratif dans lequel un groupe d'étudiants initialisés de façon aléatoire apprennent ensemble et s'entraînent mutuellement en utilisant un objectif d'apprentissage supervisé standard et un objectif de mimétisme. Ce scénario basé sur l'entraînement par les pairs favorise la convergence des modèles par rapport à un scénario d'apprentissage supervisé conventionnel Zhang *et al.* (2018).

2.3.1 *Algorithme d'apprentissage mutuel*

Zhang *et al.* (2018) proposent une procédure de distillation appelée apprentissage mutuel. Contrairement aux procédures précédentes, la distillation n'a pas lieu en sens unique mais résulte d'un jeu de rôle entre l'ensemble des modèles. L'algorithme 1 détaille la procédure d'apprentissage mutuel. Dans le cadre de l'apprentissage mutuel, les auteurs initialisent un groupe d'étudiants qui apprennent à la fois indépendamment et simultanément à résoudre une tâche donnée via la distillation de pseudo-labels.

Algorithme 1 : Procédure d'apprentissage mutuel (Zhang *et al.*, 2018)

Input : Données d'entraînement $\mathcal{X}$, ensemble des labels $\mathcal{Y}$, taux d'apprentissage γ_t, nombre de modèles K.
Initialisation : initialisation des modèles $\{\theta_1, \theta_2, .., \theta_K\}$ avec des paramètres distincts.
$t \leftarrow 0$;
tant que *convergence* **faire**
    $t \leftarrow t + 1$
    Échantillonner aléatoirement les données x, y de $\mathcal{X}, \mathcal{Y}$.
    pour $i = 0;\ i < N;\ i++$ **faire**
        $\boldsymbol{p}_i \leftarrow$ prédictions de l'étudiant θ_i
        $\mathcal{L}_i \leftarrow$ Loss supervisé$(y, \boldsymbol{p}_i)$
        pour $j = 0;\ j < N;\ j++$ **faire**
            $\boldsymbol{p}_j \leftarrow$ prédictions du professeur θ_j
            si $j \neq i$ **alors**
                $\mathcal{L}_i \leftarrow \mathcal{L}_i + KL(\boldsymbol{p}_j, \boldsymbol{p}_i)$
        Mise à jour du modèle θ_i avec la loss $\mathcal{L}_i$ et le taux d'apprentissage γ_t
Retourner : $\{\theta_1, \theta_2, .., \theta_K\}$

Les auteurs montrent que cette méthode permet d'obtenir des réseaux compacts plus performants que ceux distillés à partir d'un enseignant plus complexe mais statique.

2.3.2 *Apprentissage mutuel pour le traitement du langage naturel*

Wu *et al.* (2021) utilisent le concept d'apprentissage mutuel pour fusionner les connaissances complémentaires de modèles de langage entraîné sur des corpus distincts et via différentes procédures de pré-entraînement. Les auteurs proposent une fonction objectif triple pour l'ensemble des modèles. Le premier objectif est standard et consiste à entraîner chaque étudiant sur la tâche de modélisation du langage naturel via la procédure de langage masquée. Le second objectif consiste à distiller les probabilités en sortie des professeurs via la mesure d'entropie croisée. Le troisième objectif aligne les représentations des dernières couches des Transformers via une mesure d'erreur quadratique moyenne.

Dans le cadre de la fusion d'espace sémantique distinct, Zhao *et al.* (2021) appliquent la procédure d'apprentissage mutuel à l'entraînement coopératif d'un modèle de traduction automatique et d'un modèle de reconnaissance automatique de la parole.

3 Conclusion

Word2Vec (Mikolov *et al.*, 2013b) et Glove (Pennington *et al.*, 2014) construisent une représentation des mots à partir d'un objectif supervisé. Bojanowski *et al.* (2017) proposent une extension au modèle Word2Vec en prenant en compte la structure interne de chaque mot (n-grams).

Peters *et al.* (2018) créent le modèle ElMo et modélisent de façon profonde et contextualisée des mots d'une phrase. ElMo est un modèle de langage causal et superficiellement bi-directionnelle, lui permettant de capturer plus d'informations. Grâce à l'architecture Transformer (Vaswani *et al.*, 2017) et à la procédure de langage masqué (Devlin *et al.*, 2019), BERT parvient à modéliser le langage de façon profondément bi-directionnelle.

De nombreuses publications récentes travaille améliorent les procédures de pré-entraînement des modèles de langage. Liu *et al.* (2019) étudient les hyperparamètres associés à l'entraînement des modèles de langage masqué. Yang *et al.* (2019); Clark *et al.* (2020); Lewis *et al.* (2020a) proposent de nouvelles procédures de pré-entraînements. Radford *et al.* (2018, 2019); Brown *et al.* (2020) montrent qu'un nombre croissant de données d'entraînement et de paramètres contribuent à améliorer la capacité de généralisation du modèle de langue.

La taille croissante des modèles de langage motive les recherches dédiées à la compression des modèles via la procédure de distillation des connaissances (Tang *et al.*, 2019b; Sanh *et al.*, 2019; Kim et Rush, 2016). La distillation des représentations internes des modèles et du mécanisme d'attention favorise la convergence de l'étudiant (Sun *et al.*, 2019a; Jiao *et al.*, 2020).

Zhang *et al.* (2018) définissent un nouveau cadre pour la distillation appelé apprentissage mutuel dans lequel l'ensemble des modèles coopèrent de façon dynamique. Wu *et al.* (2021) créent un espace sémantique commun à partir de modèle de langage distinct via la procédure d'apprentissage mutuel. (Zhao *et al.*, 2021) montrent qu'un modèle de traduction automatique et un modèle de reconnaissance automatique du langage sont complémentaires lorsqu'ils sont entraînés via la procédure d'apprentissage mutuel.

Nous nous intéresserons à l'augmentation des modèles de langage via les graphes de connaissances dans les chapitres 4 et 6. Nous présenterons un nouveau cadre de distillation dédié aux graphes de connaissances dans le chapitre 5. En suite, nous présenterons une procédure originale de distillation dédiée à l'augmentation des modèles de langage dans le chapitre 6.

4

AUGMENTATION DES MODÈLES DE LANGUE PAR LA CONNAISSANCE STRUCTURÉE.

Introduction

Les modèles de langue peuvent stocker des connaissances relationnelles présentes dans les données d'apprentissage (Petroni *et al.*, 2019), en outre, ils utilisent les connaissances qu'ils ont développés lors du pré-entraînement pour résoudre des tâches extrinsèques (Cui *et al.*, 2021). Par conséquent les informations relatives aux entités disponibles dans les graphes de connaissance représentent un levier pour accroître la compréhension des modèles de langage (Qiu *et al.*, 2020; Yang *et al.*, 2021).

La nature des informations relatives aux entités disponibles dans les graphes et les méthodes dédiées à l'injection de ces connaissances font l'objet d'études approfondies (Levine *et al.*, 2020; Ke *et al.*, 2020).

Zhang *et al.* (2019e); Peters *et al.* (2019); He *et al.* (2020) modifient de façon structurelle les Transformers afin d'intégrer profondément les représentations des entités produites par les modèles de graphes. En raison de la difficulté à combiner deux espaces distincts via l'injection des embeddings des KBs au sein des modèles de langage, Weijie *et al.* (2020); Sun *et al.* (2020a) proposent d'alimenter les modèles de langue avec une représentation littérale des triplets qui sont l'unité fondamentale d'information dans un graphe de connaissance. Yamada *et al.* (2020); Wang *et al.* (2020a, 2021a); Daza *et al.* (2021) assignent au modèle de langue la responsabilité de construire des représentations des entités et des relations des graphes de connaissances via la définition d'une tâche primaire ou auxiliaire dans le cadre de l'entraînement du modèle.

Notre objectif est de comprendre les procédures d'augmentation des modèles de langage avec des graphes de connaissances et d'analyser les différents paradigmes de représentation des entités et du langage dans un espace commun.

1 Procédures d'augmentation des modèles de langue

1.1 *Augmentation par les informations des entités*

1.1.1 *SenseBERT*

Dans leur article, Levine *et al.* (2020) proposent SenseBERT, une extension du modèle BERT qui incorpore des informations provenant d'un graphe de connaissances afin d'améliorer sa capacité à comprendre la signification des mots en fontion du contexte. SenseBERT utilise les métadonnées associées aux champs lexicaux des entités du graphe de connaissances WordNet (Oram, 2001) pour fournir une abstraction de la signification des mots dans une phrase.

Les auteurs définissent une fonction objective auxiliaire pendant le pré-entraînement de SenseBERT qui vise à prédire la classe sémantique d'un mot masqué en fonction de son contexte. En particulière, l'utilisation des corpus textuels avec des alignements entre le texte et les sens sont utilisés. Finalement, 45 champs lexicaux sont sélectionnés pour évaluation dans la tâche de prédiction de similarité sémantique.

SenseBERT dispose de trois objectifs : 1) l'objectif de pré-entraînement standard associé aux modèles de langage masqué $\mathcal{L}_{MLM}$; 2) un objectif dédié à la modélisation des champs lexicaux $\mathcal{L}_{SLM}$; 3) un objectif de régularisation $\mathcal{L}_{REG}$.

$$
\begin{aligned}
\mathcal{L}_{MLM} &= -\log p(w_t \mid context) \\
\mathcal{L}_{SLM} &= -\log \sum_{s \in A(w_t)} p(s \mid context) \\
\mathcal{L}_{REG} &= -\sum_{s \in A(w_t)} \frac{1}{|A(w_t)|} \log p(s \mid context) \\
\mathcal{L}_{SSB} &= \mathcal{L}_{MLM} + \mathcal{L}_{SLM} + \mathcal{L}_{REG}
\end{aligned}
\tag{4.1}
$$

La fonction $A(w_t)$ retourne l'ensemble des champs lexicaux associé au mot w_t.

Le modèle SenseBERT représente une avancée significative dans la capacité des modèles de langage à comprendre le langage naturel en exploitant les informations d'un graphe de connaissances. L'utilisation des champs lexicaux comme objectif auxiliaire pendant le pré-entraînement permet au modèle d'apprendre une représentation plus abstraite de la signification des mots, améliorant ainsi ses performances sur une variété de tâches de traitement du langage naturel.

1.1.2 *SentiLARE*

Ke *et al.* (2020) présentent un modèle pré-entraîné appelé SentiLARE pour l'analyse des sentiments qui incorpore les connaissances linguistiques du graphe SentiWordNet (Baccianella *et al.*, 2010) à travers un mécanisme d'attention contextuelle. SentiLARE a deux couches d'intégration : une pour le sentiment des mots individuels, et une pour le sentiment de la phrase entière. Le modèle est pré-entraîné par un processus en deux étapes appelé fusion précoce et tardive des connaissances. Dans la phase initiale, le modèle est entraîné au niveau des mots, tandis que dans la phase finale, il est entraîné au niveau de la phrase et des mots.

Le modèle SentiLARE dispose de deux couches d'embeddings. La première couche encode le sentiment des mots individuels en utilisant les informations de la base de connaissances SentiWordNet, tandis que la seconde couche encode le sentiment de la phrase entière. Pendant la phase initiale de la fusion de connaissances, les deux couches sont utilisées pour entraîner le modèle, tandis que pendant la phase finale, seul le sentiment des mots individuels est incorporé. Cela permet au modèle d'apprendre à la fois des informations locales et globales sur les sentiments de la phrase.

Objectif de fusion anticipée des connaissances :

$$\begin{aligned}\mathcal{L}_{EF} = -\sum_{t=1}^{n} m_t \cdot [\log P(x_t \mid X', l) + .. \log P(\text{pos}_t \mid X', l) + \\ \log P(\text{polar}_t \mid X', l)]\end{aligned} \tag{4.2}$$

m_t est une fonction indicatrice qui est égale à 1 si le mot est masqué et 0 sinon conformément à la procédure de MLM. X' désigne la séquence de textes enrichis en connaissances. pos_t est un label obtenu via une procédure d'étiquetage morphosyntaxique. $polar_t$ est le sentiment associé au mot t.

Objectif de fusion tardive des connaissances :

$$\mathcal{L}_{LS} = -\log P(l \mid X') - \sum_{t=1}^{n} m_t \cdot [\log P(x_t \mid X') + .. \log P(\text{pos}_t \mid X') + \log P(\text{polar}_t \mid X')] \tag{4.3}$$

Avec le label l désigne le sentiment associé à la séquence X.

1.2 *Intégration des représentations des modèles de graphes*

1.2.1 *ERNIE*

ERNIE (Zhang *et al.*, 2019d) est un modèle de langage masqué qui propose un module d'intégration des embeddings des entités. Ce modèle étend le pré-entraînement de BERT en introduisant un nouvel objectif, à savoir retrouver des entités masquées dans un texte. L'architecture de ERNIE se compose d'un encodeur textuel, qui capture les informations lexicales et syntaxiques du contexte, et d'un encodeur de connaissances, qui intègre les embeddings d'entités aux représentations textuelles.

Le module de fusion du K-Encoder permet d'agréger les représentations d'entités et de contexte, et plusieurs modules K-Encoder peuvent être empilés pour construire un espace commun partagé par les représentations des entités et du texte.

1.2.2 *KnowBERT*

Le module KAR de Peters *et al.* (2019) utilise une matrice de projection $\mathbf{W_1^{\text{proj}}}$ pour projeter les représentations de chaque mot dans une dimension de représentation des entités. Cette projection est donnée par l'équation suivante :

$$\mathbf{H}i^{\text{proj}} = \mathbf{H}i\mathbf{W1}^{\text{proj}} + \mathbf{b}1^{\text{proj}}. \tag{4.4}$$

Les représentations de mentions d'entités, notées $\mathbf{H}_i^{\text{proj}}$, sont concaténées dans une matrice $\mathbf{S} \in \mathbb{R}^{C \times E}$, où C est le nombre de mentions d'entités et E est la dimension des représentations d'entités dans le modèle de graphe. Si une mention est composée de plusieurs mots, les auteurs appliquent une opération de pooling sur ces représentations.

Le module KAR utilise une opération d'auto-attention sur les représentations des mentions pour intégrer les interactions potentielles entre elles. Cette opération est notée $\mathbf{S}^e = \text{Auto-attention}(\mathbf{S})$.

Après avoir contextualisé les représentations des mentions, KnowBERT enrichit ces représentations en utilisant les embeddings des entités du graphe de connaissances. Cette opération est donnée par l'équation suivante :

$$\mathbf{s}'^e_m = \mathbf{s}^e_m + \sum_t \text{MLP}(p_{mt}, \mathbf{s}^e_m \cdot \mathbf{e}_{mt})\mathbf{e}_{mt} \tag{4.5}$$

Dans cette équation, t désigne le nombre d'entités susceptibles de correspondre à la mention m, p_{mt} est la probabilité a priori que l'entité e_{mt} corresponde à la mention m, et $\mathbf{e}_{mt} \in \mathbb{R}^k$ est la représentation de l'entité e_{mt} calculée à partir d'un modèle de graphe. La fonction MLP est un perceptron multicouches chargé de pondérer les embeddings des entités candidates pour la mention m.

Ensuite, l'intégration des annotations sémantiques consiste à concaténer les représentations enrichies des mentions dans une matrice $\mathbf{S}'^e \in \mathbb{R}^{C \times E}$. Cette étape permet de regrouper les représentations des mentions pour faciliter la recontextualisation par KnowBERT. Ce dernier utilise une couche de type Transformer modifiée qui remplace l'auto-attention par une attention entre les représentations des mots et des entités. Cela peut être exprimé mathématiquement comme suit :

$$\mathbf{H}i'^{proj} = \text{MLP}(\text{ MultiHeadAttention}(\mathbf{H}i^{\text{proj}}, \mathbf{S}'e, \mathbf{S}'e)) \tag{4.6}$$

Enfin, la représentation $\mathbf{H}'^{proj}_i$ est projetée dans la dimension de BERT en utilisant une transformation linéaire, comme suit :

$$\mathbf{H}i' = \mathbf{H}i'^{\text{proj}}\mathbf{W}2^{\text{proj}} + \mathbf{b}2^{\text{proj}} + \mathbf{H}_i \tag{4.7}$$

La connexion résiduelle $\mathbf{H}_i$ est également utilisée.

1.2.3 *BERT-MK*

BERT-MK (He *et al.*, 2020) est une extension du modèle ERNIE et construit des représentations contextualisées des entités. Le modèle se compose de deux

modules : un module d'apprentissage des connaissances et un module de pré-entraînement du modèle de langage. Le premier module est utilisé pour apprendre les représentations contextualisées des entités à partir d'un graphe de connaissances, tandis que le second intègre ces connaissances dans le modèle de langage.

Pour chaque entité du graphe de connaissance, les auteurs construisent un sous-graphe composé des triplets associés à l'entité et des triplets de son voisinage, qui peuvent être représentés comme suit : $(e_1, r_1, e), (e, r_2, e_2), (e, r_3, e_3), (e_4, r_4, e)$. Les auteurs considèrent ensuite les entités et les relations comme une séquence et les convertissent en représentations via une matrice d'embeddings. Ces représentations sont alors utilisées pour entraîner un Transformer dont l'objectif est de reconstruire les triplets en entrée du modèle.

Fonction objectif de reconstruction des triplets (Han *et al.*, 2018) :

$$\mathcal{L}_{triplets} = \sum_{(\mathbf{h},\mathbf{r},\mathbf{t}) \in T^{+}_{batch}} \max\{|\mathbf{h} + \mathbf{r} - \mathbf{t}| - |\mathbf{h}' + \mathbf{r} - \mathbf{t}'| + \gamma, 0\} \tag{4.8}$$

Avec $\mathbf{h}$, $\mathbf{r}$, $\mathbf{t} \in \mathbb{R}^k$, les représentations des entités et des relations produites par le Transformer. T^{+}_{batch} est l'ensemble des triplets positifs de la séquence.

1.3 *Représentation littérale des triplets*

1.3.1 *K-BERT*

Les représentations des entités et du texte sont généralement produites par des modèles distincts. Face à la difficulté d'aligner les représentations des modèles de graphe et de texte Weijie *et al.* (2020) proposent d'augmenter les Transformers en représentant les triplets de façon littérale au sein du texte.

Le modèle K-BERT propose une extension à la couche d'embeddings des Transformers. Via la création d'un arbre, K-BERT attribue à un tuple {relation, entité} la position qui suit la mention d'une entité (soft positions). La somme des embeddings de segmentation, de position et des mots forment la représentation du mot responsable d'alimenter le Transformer. Les embeddings de segmentation sont un marqueur pour les phrases.

Les auteurs proposent un module original appelé matrice de visibilité afin d'éviter d'incorporer trop d'informations et risquer de détourner la phrase de son sens original. La matrice de visibilité intervient dans le mécanisme d'attention en re-

streignant la visibilité des entités et des relations ajoutées lors la procédure d'augmentation à la mention auxquels elles sont connectées.

Les résultats empiriques du modèle K-BERT montrent qu'il est performant pour les tâches liées à un domaine spécifique et axées sur la connaissance et peut être utilisé pour résoudre des problèmes qui nécessitent des experts du domaine.

1.3.2 *CoLAKE*

Le modèle CoLAKE (Sun *et al.*, 2020a) intègre le contexte de la langue et le contexte de la connaissance dans une structure de données de façon similaire à K-BERT (voir section 1.3.1). Pour mettre en œuvre l'apprentissage conjoint du langage et des connaissances, les auteurs proposent une extension à la procédure de MLM et l'appliquent aux triplets du graphe de connaissance.

Les auteurs identifient les mentions d'entités dans le texte et étendent la séquence en entrer du modèle à partir des triplets qui sont connectés aux mentions. Ils ne considèrent que les triplets dans lesquels la mention correspond à la tête (sujet) au lieu de la queue (objet) du triplet. De façon similaire à K-BERT, les auteurs intègrent une matrice de visibilité au sein du mécanisme d'attention qui restreint la visibilité des entités et des relations ajoutées lors la procédure d'augmentation. La procédure de pré-entraînement de CoLAKE étend la procédure de MLM de BERT et entraîne le modèle à retrouver les mots, les relations et les entités masqués.

Les auteurs démontrent l'efficacité du modèle CoLAKE sur les tâches exigeant des connaissances via les résultats expérimentaux.

1.4 *Apprentissage des entités via le Transformer et définition d'une fonction objective*

La section suivante présente des modèles de langage profond qui développent une représentation interne des entités via la définition d'une fonction objective et de paramètres spécifiques.

1.4.1 *LUKE*

Yamada *et al.* (2020) créent le modèle LUKE et proposent une extension du mécanisme d'attention permettant de réaliser la distinction entre les mots et les entités lors du calcul des scores d'attention. Les auteurs également proposent une nouvelle tâche de pré-entraînement basée sur la procédure de MLM.

Le mécanisme d'auto-attention met en relation les mots entre eux sur la base du score d'attention entre chaque paire de mots. Étant donné une séquence de vecteurs d'entrée $\{\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, .., \mathbf{x}_k\}$ ou $x_i \in \mathbb{R}^D$, le mécanisme d'auto-attention produit des scores d'attention e_{ij}. Le modèle LUKE définit *l'auto-attention consciente des entités* et initialise trois nouvelles matrices dédiées aux entités dans le cadre du calcul des scores d'attentions : $\mathbf{Q}_{w2e}, \mathbf{Q}_{e2w}, \mathbf{Q}_{e2e} \in \mathbb{R}^{L \times D}$.

Calcul des scores d'attention par le modèle LUKE :

$$\mathbf{y}_i = \sum_{j=1}^{k} \alpha_{ij} \mathbf{V} \mathbf{x}_j$$

$$e_{ij} = \begin{cases} \mathbf{K}\mathbf{x}_j^\top \mathbf{Q}\mathbf{x}_i, & \text{si } \mathbf{x}_i \text{ et } \mathbf{x}_j \text{ sont des mots} \\ \mathbf{K}\mathbf{x}_j^\top \mathbf{Q}_{w2e}\mathbf{x}_i, & \text{si } \mathbf{x}_i \text{ est un mot et } \mathbf{x}_j \text{ est une entité} \\ \mathbf{K}\mathbf{x}_j^\top \mathbf{Q}_e 2w\mathbf{x}_i, & \text{si } \mathbf{x}_i \text{ est une entité et } \mathbf{x}_j \text{ est un mot} \\ \mathbf{K}\mathbf{x}_j^\top \mathbf{Q}_e 2e\mathbf{x}_i, & \text{si } \mathbf{x}_i \text{ et } \mathbf{x}_j \text{ sont des entités} \end{cases} \tag{4.9}$$

$$\alpha_{ij} = \text{softmax}(e_{ij})$$

Pour construire des poids pertinents associés aux matrices $\{\mathbf{Q}_{w2e}, \mathbf{Q}_{e2w}, \mathbf{Q}_{e2e}\}$, le modèle LUKE étend la procédure de MLM et masque aléatoirement des entités pour entraîner le modèle à les retrouver.

Les résultats expérimentaux montrent l'efficacité du modèle sur diverses tâches liées aux entités comme sur la tâche questions-réponses, de classification des relations et de reconnaissance d'entités nommées.

1.4.2 *K-ADAPTER*

L'intégration de nouvelles connaissances au sein d'un modèle pré-entraîné présente des risques d'oubli catastrophique (French, 1999). Pour y remédier, (Wang *et al.*, 2020a) proposent via le modèle K-ADAPTER de conserver les poids originaux de RoBERTa fixé et d'infuser la connaissance de RoBERTa au sein d'un modèle qui joue le rôle d'adaptateur.

Les auteurs concatènent les représentations en sortie de chaque couche du PLM (RoBERTa) avec celles de l'adaptateur pour alimenter la couche suivante de l'adap-

tateur. Les représentations finales des deux modèles sont concaténées afin d'alimenter le module de classification de l'adaptateur.

En outre, les auteurs proposent une procédure appelée *adaptateur factuel* qui considère la tâche de classification des relations entre les entités comme un objectif auxiliaire de pré-entraînement. Cette procédure vise à injecter des connaissances factuelles au sein de l'adaptateur. La procédure K-ADAPTER associé au pré-entraînement sur la tâche de classification de relations permet au modèle de conserver les connaissances apprises lors de son pré-entraînement tout en développant ses connaissances factuelles sur les entités.

1.4.3 *KEPLER*

Le modèle KEPLER (Wang *et al.*, 2021a) construit des représentations des entités et du texte dans un espace commun via la définition d'un objectif multitâche. La procédure de pré-entraînement du modèle KEPLER se décompose en deux objectifs. Le premier est la tache de langage masquée et le second est la tâche de prédiction des liens.

Les auteurs remplacent les identifiants des entités par leurs descriptions textuelles dans l'objectif de produire des représentations des entités. KEPLER dispose d'une matrice d'embeddings externe dédiée au traitement des relations. Le modèle intègre la fonction de score de TransE pour produire un score de vraisemblance à partir d'un triplet.

Fonction objectif Sun *et al.* (2019b) associée à la tâche de prédiction des liens :

$$\mathcal{L}_{\text{PL}} = -\log \sigma(\gamma - \|\mathbf{h} + \mathbf{r} - \mathbf{t}\|) - \sum_{i=1}^{k} \frac{1}{k} \log \sigma(\|\mathbf{h}' + \mathbf{r} - \mathbf{t}'\| - \gamma) \tag{4.10}$$

Avec $\mathbf{h}$ et $\mathbf{t} \in \mathbb{R}^k$ les représentations des descriptions des entités. $\mathbf{h}', \mathbf{t}' \in \mathbb{R}^k$ correspondent aux représentations de l'objet et du prédicat d'un triplet corrompu.

L'objectif de prédiction des liens renforce la capacité du modèle à extraire des connaissances du texte puisqu'il exige que KEPLER encode les entités à partir de leurs descriptions correspondantes.

Daza *et al.* (2021) proposent le modèle BLP comme une extension de KEPLER. Le modèle BLP conserve que l'objectif de prédiction de liens de KEPLER et supprime l'objectif de modélisation du langage masqué. La procédure d'entraînement de BLP permet de construire un Transformer adapté à la recherche d'entité.

2 Conclusion

Apprendre les représentations du langage et des connaissances dans un espace commun améliore la capacité des modèles de langue sur une variété d'applications. Les recherches récentes en matière d'augmentation des modèles de langage travaillent à développer des procédures robustes.

Levine *et al.* (2020) augmentent les capacités sémantiques et lexicales des Transformers. Les auteurs étendent la procédure de MLM via un objectif auxiliaire qui consiste à prédire les champs lexicaux. Pour augmenter les capacités lexicales des Transformer Ke *et al.* (2020) intègrent les sentiments associés aux mots et à la phrase via la couche d'embeddings des Transformers. En associant l'augmentation à la définition d'une tâche auxiliaire de prédiction des sentiments, les auteurs parviennent à créer un modèle qui capture mieux la polarité des sentiments d'une phrase.

Zhang *et al.* (2019d); Peters *et al.* (2019) définissent des modules permettant d'intégrer profondément les représentations des modèles de graphes au sein des modèles de langue. Zhang *et al.* (2019d) créent le K-Encoder qui est une extension du mécanisme d'attention permettant d'intégrer les représentations de TransE. He *et al.* (2020) développent un modèle capable de construire des représentations contextualisées des entités et les injectent dans le K-Encoder de ERNIE. Peters *et al.* (2019) propose le module KAR qui produit des scores d'attentions qui tiennent compte des différentes représentations probables d'une mention d'entité et des interactions entre les mentions d'entités au sein des phrases.

Face à la difficulté d'alignement des représentations des graphes et du texte, Weijie *et al.* (2020); Sun *et al.* (2020a) proposent d'intégrer dans le texte les triplets associés aux entités mentionnées. Pour ce faire ils modifient les couches d'embeddings des Transformers pour initialiser des représentations associées aux entités et aux relations qu'ils traitent comme des mots. Les auteurs proposent de régulariser l'intégration des connaissances pour conserver le sens original de la phrase via des matrices de visibilités limitant ainsi la portée des triplets dans le mécanisme d'attention.

Yamada *et al.* (2020); Wang *et al.* (2020a, 2021a); Daza *et al.* (2021) attribuent aux Transformers la responsabilité de créer des représentations des entités. Ils développent des architectures de modèles et des fonctions objectives dédiées aux entités. Yamada *et al.* (2020) modifient le mécanisme d'attention en attribuant des poids spécifiques aux entités. Les auteurs étendent la procédure de MLM aux entités pour initialiser les poids correspondants. Wang *et al.* (2020a) créent les adaptateurs pour pallier à l'oubli des connaissances lors de l'intégration de connaissances

structuré (French, 1999). L'adaptateur est un modèle spécialisé sur une tâche extrinsèque dont les couches reçoivent les représentations des couches d'un modèle figé. Wang *et al.* (2021a) intègrent la connaissance des entités en définissent la tâche de prédiction des liens comme un objectif auxiliaire lors du pré-entraînement du modèle.

Partie II

CONTRIBUTIONS

5

APPRENTISSAGE DE REPRÉSENTATIONS DE GRAPHES DE CONNAISSANCES PAR DISTILLATION COOPÉRATIVE

1 Contexte et motivations

Bien que les bases de connaissances classiques puissent inclure une quantité importante de connaissances observées à travers des millions d'entités et leurs relations, elles sont par nature incomplètes puisqu'elles ne peuvent capturer qu'une fraction des connaissances du monde. Cette limitation a donné lieu à de nombreux travaux de recherche axés sur la prédiction de nouvelles connaissances à partir des connaissances existantes des KBs. Cette question a été abordée avec succès par des approches neuronales pour l'apprentissage de la représentation des KBs (Wang *et al.*, 2017; Sun *et al.*, 2019b; Bordes *et al.*, 2013; Yang *et al.*, 2015b). Ces modèles visent à représenter les entités et les relations de la base de connaissances dans des espaces de faible dimension pour inférer les relations entre les entités. Ces dernières années ont été marquées par un intérêt croissant pour les modèles de représentation utilisés pour connecter plusieurs bases de connaissances. Une révision détaillé de la littérature est présenté dans la section 2.2.

L'objectif principal de l'apprentissage de la représentation multigraphes est d'habiliter les modèles d'entités et de relations avec différents contextes de graphes qui peuvent potentiellement faire le lien avec différents contextes sémantiques. Pour atteindre cet objectif, des embeddings sont appris sur les triplets combinés à travers les graphes. Bien que les méthodes d'apprentissage de la représentation multigraphe aient obtenu des résultats prometteurs, elles sont toujours confrontées à deux limitations principales. Tout d'abord, elles sont particulièrement adaptées à l'alignement des graphes et à la traduction automatique en tant que tâches extrinsèques et entraînent des défis de passage à l'échelle dans les KBs à grande échelle. Deuxièmement, ces méthodes supposent que chaque base de connaissance a accès à toutes les entités et relations stockées dans les autres bases de connaissance, alors

qu'il n'est peut-être pas possible ni pertinent pour les bases de connaissances de partager des informations non alignées, comme dans les bases de connaissances personnelles (Balog et Kenter, 2019).

En suivant un objectif différent, nous soutenons qu'en dehors de toute tâche extrinsèque, la modélisation des schémas relationnels entre les KBs pourrait principalement se concentrer sur la modélisation explicite des connexions entre les entités au sein de chaque KB en utilisant ses propres triples observés et, inférer les schémas supplémentaires de son pair en utilisant uniquement des triples partiellement alignés. À titre d'exemple, considérons deux bases de connaissances (KB^1 et KB^2) qui contiennent des faits concernant des villes, des capitales et des pays, comme l'illustre la figure 5.1, mais dont aucune n'inclut le fait que *Rome est une ville d'Italie*. Via l'apprentissage de l'espace de la base de connaissances KB^1, le modèle associé peut être capable de généraliser correctement la relation *CityIn* et déduire que *Rome est une ville d'Italie* en regroupant d'autres représentations d'entités similaires à *Rome* telles que l'embedding de *Pisa*.

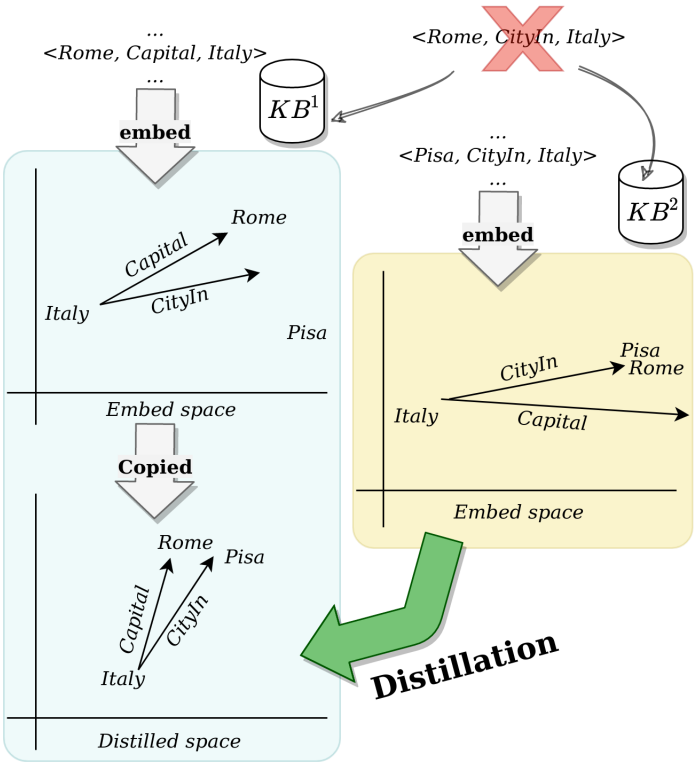


FIGURE 5.1 – Procédure de prédiction des champs lexicaux illustrée.

Bien que cette connaissance n'ait pas été inférée par KB^1, KB^2 peut enseigner cette information à KB^1 par distillation. Alors, le modèle de KB^1 sera capable de

comprendre la relation *CapitalOf* en observant directement des exemples dans son propre contexte sémantique et la relation *CityIn* par la connaissance distillée du contexte sémantique de KB^2.

En conséquence, contrairement aux travaux existants sur l'intégration multi-graphes qui reposent sur une vue unifiée de plusieurs graphes, notre travail repose plutôt sur de multiples vues au sein des KBs qui sont reliées par des informations alignées. Alors que chaque KB peut apprendre des représentations sur son propre contexte sémantique basé sur des triplets associés, il peut en outre échanger des connaissances inférées à partir de triplets alignés fournis par d'autres GC, améliorant à leur tour les représentations des autres KB sur la base de contextes sémantiques différents. Notre idée principale est de modéliser un processus de distillation des connaissances (Hinton *et al.*, 2015a) à travers les KBs pour renforcer leur capacité de généralisation.

Malgré le nombre de travaux qui étudient l'inférence d'entités et de relations entre les bases de connaissances, aucun n'a montré la faisabilité du cadre de distillation des connaissances pour modéliser l'inférence de connaissances entre les bases de connaissances. Puisque les bases de connaissances jouent des rôles symétriques dans le transfert de connaissances, une question critique est de savoir comment entraîner chaque modèle de base de connaissances à l'aide des labels des entités/relations basées sur des distributions estimées par le professeur, ainsi que sur sa propre distribution estimée. Pour résoudre ce problème, nous plaidons en faveur d'un paradigme d'apprentissage mutuel (Zhang *et al.*, 2018), où chaque KB agit dynamiquement comme un professeur ou un étudiant. Contrairement au traditionnel transfert de connaissances statique à sens unique d'un modèle qui joue le rôle de professeur à un modèle étudiant, nous préconisons un transfert de connaissances coopératif à deux sens entre un KB et ses pairs.

Concrètement, notre configuration est la suivante : le modèle d'apprentissage de représentation de chaque KB possède deux fonctions objectives qui sont optimisées conjointement : 1) une fonction objective classique basée sur l'évaluation de la vraisemblance d'un triplet dont l'objectif est de produire des scores supérieurs pour les triples existants par rapport à des triplets corrompus ; et 2) une fonction de distillation coopérative par mimétisme qui rend les prédictions de classe postérieures des entités alignées et des relations alignées proches des probabilités de classe des entités et des relations de son pair, respectivement.

Grâce à l'optimisation conjointe, les connaissances sont également transférées naturellement, c'est-à-dire, de l'information alignée à l'information non alignée.

En résumé, les principales contributions de ce chapitre sont les suivantes : 1) une première tentative visant à mettre en place l'apprentissage de représentation mul-

tigraphe dans un cadre théorique de distillation des connaissances ; 2) un nouveau modèle d'apprentissage de la représentation des KBs appelé KD-MKB, basé sur une stratégie coopérative de distillation des connaissances ; 3) des expériences sur deux ensembles de données standards, WN18RR et FB15K-237, qui valident empiriquement le raisonnement de la distillation des connaissances à travers les KBs et montrent l'efficacité de la distillation coopérative des connaissances telle que proposée dans *KD-MKB*.

2 Problématiques et définitions

Dans cette section, nous décrivons *KD-MKB*, un modèle d'apprentissage de représentation de KB. Nous introduisons d'abord quelques définitions terminologiques afin de pouvoir définir formellement notre modèle.

2.1 *Concepts et définitions*

2.1.1 *Bases de connaissances, entités, relations et triplets*

Pour rappel, une base de connaissances KB représente un graphe $(\mathcal{E}, \mathcal{R})$ qui comprend un ensemble d'entités $\mathcal{E} = \{e_1, e_2, \dots e_{N_e}\}$, un ensemble de relations $\mathcal{R} = \{r_1, r_2, \dots r_{N_r}\}$, et un ensemble de faits réels sous forme de triples positifs (e_x, r_w, e_y) noté T^+ parmi tous les triplets possibles dans $\mathcal{E} \times \mathcal{R} \times \mathcal{E}$. L'ensemble des triplets négatifs est noté T^-.

2.1.2 *Alignement des entités et des relations*

Soit $\mathcal{KB} = \{KB^1, KB^2, \dots KB^n\}$ représentant une collection de KBs. Pour une paire $(KB^i, KB^j) \in \mathcal{KB}^2$, KB^i (resp. KB^j) comprend un ensemble d'entités $\mathcal{E}^i$ (resp. $\mathcal{E}^j$), un ensemble de relations $\mathcal{R}^i$ (resp. $\mathcal{R}^j$). $I_e(i,j) = \{(e^i_x, e^j_y) \in \mathcal{E}^i \times \mathcal{E}^j\}$ est l'ensemble des entités alignées signifiant que e^i_x et e^j_y représentent la même entité du monde réel.

Notez que l'ensemble des entités de KB^i, noté $I^i_e(i,j)$, est égal à sa contrepartie $I^j_e(i,j)$, et $|I_e(i,j)| = |I^i_e(i,j)|$. De même, $I_r(i,j) = \{(r^i_v, r^j_w) \in \mathcal{R}^i \times \mathcal{R}^j\}$ désigne l'ensemble des relations alignées (r^i_v, r^j_w) entre KB^i et KB^j telles que r^i_v et r^j_w représentent des relations équivalentes, et $|I_r(i,j)| = |I^i_r(i,j)|$.

Enfin, soit $I^i_t(i,j) = \{(e^i_{x1}, r^i, e^i_{x2}) : \forall e^i_{x1}, e^i_{x2} \in I^i_e(i,j), \forall r^i \in I^i_r(i,j)\}$ et $I^j_t(i,j) = \{(e^j_{x1}, r^j, e^j_{x2}) : \forall e^j_{x1}, e^j_{x2} \in I^j_e(i,j), \forall r^j \in I^j_r(i,j)\}$ sont les ensembles de triplets possibles formés à partir d'entités et de relations alignées pour chaque KB.

2.2 *Définition du problème*

Nous pensons que le transfert de connaissances entre les bases de connaissances peut être réalisé par la distillation des relations et des entités. Dans ce qui suit,

nous exposons nos intuitions derrière la distillation des relations et des entités entre les bases de connaissances, comme illustré dans la figure 5.2.

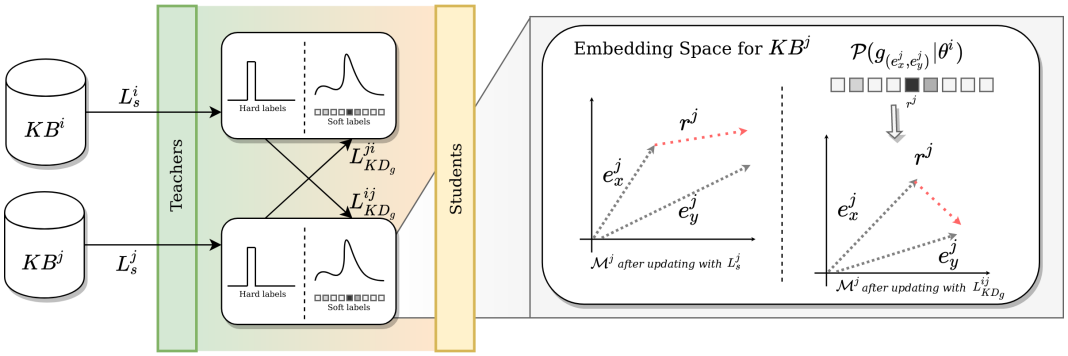


FIGURE 5.2 – Architecture du modèle KD-MKB. Le zoom sur le modèle $\mathcal{M}^j$ est illustré par un exemple de distillation de relations.

2.2.1 *Distillation des relations*

Considérons (e_1^i, e_x^j), $(e_2^i, e_y^j) \in I_e(i,j)$ deux paires d'entités alignées entre KB^i et KB^j. En supposant l'existence de relations alignées entre KB^i et KB^j, notre intuition est que de telles paires d'entités conduisent à la même probabilité d'inférence de relation car les entités alignées font référence aux mêmes objets du monde réel Sun *et al.* (2018); Zhu *et al.* (2017a). En conséquence, nous soutenons l'idée de distiller mutuellement les relations alignées probables d'une KB à ses pairs. Formellement, les scores de plausibilité des triplets (e_1^i, r^i, e_2^i) peuvent être estimés avec une confiance élevée sur la base des scores de plausibilité des triplets (e_x^j, r^j, e_y^j) et vice versa.

2.2.2 *Distillation des entités*

Considérons $(r_v^i, r_w^j) \in I_r(i,j)$ et $(e_1^i, e_x^j) \in I_e(i,j)$ une paire d'entités alignées et de relations entre KB^i et KB^j. impliquée dans le triplet $(e_1^i, r_v^i, e_2^i), \in T_i^+$. De manière similaire à la distillation des relations, nous pensons que de telles paires de relations conduisent à la même probabilité d'inférence d'entité car les relations alignées apportent une sémantique équivalente qui relie les entités Sun *et al.* (2018); Zhu *et al.* (2017a). Ainsi, nous soutenons la pertinence de distiller mutuellement les entités alignées probables d'une KB à son homologue. Par analogie au principe de distillation des relations, les scores de plausibilité des triplets (e_1^i, r_v^i, e_2^i) peuvent être estimés avec une confiance élevée sur la base des scores de plausibilité des triplets (e_x^j, r_w^j, e_y^j) et vice versa.

Nous étudions l'apprentissage de la représentation d'entités et de relations à travers plusieurs KB, tout en préservant l'information essentielle incluse dans chaque KB. Formellement, étant donné une collection de KBs $\mathcal{KB} = \{KB^1, KB^2, \dots KB^n\}$, un modèle de représentation des connaissances $\mathcal{M}^i$ est entraîner pour préserver les entités et les relations de chaque $KB^i, i = 1 \dots n$ dans un espace de représentation séparé.

3 Formulation du modèle KD-MKB

3.1 *Principaux objectifs*

Notre objectif principal est, d'une part, d'apprendre les représentations directement à partir des connaissances incluses dans chaque KB, et d'autre part, d'améliorer l'apprentissage en utilisant les connaissances distillées à partir de ses pairs en ce qui concerne les entités alignées et les relations alignées. Sur la base de ce principe, le cadre d'apprentissage atteint conjointement deux objectifs complémentaires.

3.1.1 *Objectif O1. Préserver la structure relationnelle de chaque KB*

Pour chaque participant KB^i, un modèle de représentation de connaissances dédié $\mathcal{M}^i$ prend des triplets (e^i_x, r^i, e^i_y) qui sont soit positifs dans T^+_i, soit négatifs dans T^-_i et apprend les vecteurs d'embeddings correspondants $(\mathbf{e^i_x}, \mathbf{r^i}, \mathbf{e^i_y})$ en maximisant une fonction de score de plausibilité triple $f_i : \mathcal{E}^i \times \mathcal{R}^i \times \mathcal{E}^i$ dans un espace de dimension k_i. Des exemples de fonctions de score de l'état de l'art comme TransE (Bordes *et al.*, 2013), TransH (Wang *et al.*, 2014c) et RotatE (Sun *et al.*, 2019b) sont fournis dans la section 3.2.1.

3.1.2 *Objectif O2. Améliorer la capacité de généralisation du modèle d'apprentissage de représentation de chaque KB en s'appuyant sur ses pairs*

Dans un contexte d'apprentissage coopératif, chaque modèle d'intégration de connaissances $\mathcal{M}^i$ est amélioré en utilisant les connaissances distillées par chacun des autres modèles d'intégration $\mathcal{M}^j, j = 1 \dots n, j \neq i$. Chaque modèle de KB $\mathcal{M}^i$ agit dynamiquement comme un enseignant ou un élève en distillant ou en tirant parti respectivement des relations distillées et des entités distillées de ses pairs.

Ainsi, nous formulons le modèle KD-MKB avec un ensemble de n réseaux qui agissent dynamiquement comme des réseaux d'enseignants ou d'élèves et apprennent mutuellement chacun des modèles spécifiques $\mathcal{M}^i, i = 1 \dots n$. La figure 5.2 donne un aperçu de l'architecture KD-MKB avec un ensemble de 2 ($n = 2$) enseignants étudiants.

Chaque modèle de KB $\mathcal{M}^i$ utilise un cadre enseignant élève qui apprend à partir de la vérité terrain en utilisant une fonction de score qui mesure la vraisemblance

des embeddings et des softs labels fournis par les $n-1$ prédictions des modèles enseignants sur le modèle de la distillation des relations et des entités. La distribution de probabilité associée à chaque prédiction fournie par les autres KBs enseignants $KB^j, j = 1 \dots n, j \neq i$ permet au modèle $\mathcal{M}^i$ d'apprendre des informations contextuelles plus riches sur la similarité des embeddings de relations et d'entités, conduisant à une capacité de généralisation accrue. Ainsi, le modèle $\mathcal{M}^i$ suit deux fonctions objectives qui sont optimisées conjointement : une fonction objective classique de KB supervisée L_s^i sur les labels de la vérité terrain et une fonction objectif de distillation de connaissance coopérative mimétique L_{KD}^i sur les softs labels.

$$\mathcal{L}(\theta^i) = (1-\alpha)L_s^i + \alpha L_{KD}^i \tag{5.1}$$

où α est un hyperparamètre dédié à la pondération des objectifs de l'apprentissage.

Par conséquent, chaque modèle de KB apprend à la fois à prédire correctement le label correct sur la base des triplets d'apprentissage (perte L_s^i) et à reproduire l'estimation de la probabilité postérieure des relations et des entités fournie par ses pairs (perte L_{KD}^i), selon les intuitions décrites ci-dessus. Un tel apprentissage mutuel aide chaque KB à apprendre un complément de contexte de la part de ses pairs.

3.2 *Fonctions objectifs*

3.2.1 *Fonction objectif supervisée de classification*

Conformément à l'objectif O1 (voir la section 3.1.1), nous adoptons un modèle de représentation de KB standard, à savoir TransE (Bordes *et al.*, 2013). Il convient de mentionner que d'autres modèles de représentation de KB peuvent également être utilisés (par exemple, TransH (Wang *et al.*, 2014c) ou RotatE (Sun *et al.*, 2019b)). Étant donné un fait de relation (e_x^i, r_w^i, e_y^i) dans KB^i, nous utilisons la fonction de score suivante pour estimer la plausibilité d'un triplet :

$$f_i(\mathbf{e_x^i}, \mathbf{r_w^i}, \mathbf{e_y^i}) = -\parallel \mathbf{e_x^i} + \mathbf{r_w^i} - \mathbf{e_y^i} \parallel \tag{5.2}$$

où $\parallel \cdot \parallel$ désigne la norme vectorielle L_1 ou L_2. Par conséquent, nous définissons, la probabilité que (e_x^i, r_w^i, e_y^i) soit un vrai triplet comme :

$$\mathcal{P}(y_{(e_x^i, r_w^i, e_y^i)} = 1 \mid \theta^i) = sigmoid(f_i(\mathbf{e_x^i}, \mathbf{r_w^i}, \mathbf{e_x^i})) \tag{5.3}$$

où $y_{(e_x^i, r_w^i, e_y^i)}$ est une variable aléatoire de valeur 1 si le triple (e_x^i, r_w^i, e_y^i) est vrai (c'est-à-dire une relation factuelle), et 0 sinon $sigmoid(f_i(\mathbf{e_x^i}, \mathbf{r_w^i}, \mathbf{e_x^i})) = \frac{1}{1+ \exp(-f_i(\mathbf{e_x^i}, \mathbf{r_w^i}, \mathbf{e_x^i}))}$, la fonction sigmoïde appliquée à chaque score des triplets. Les paramètres du modèle de représentation θ^i, sont définis en minimisant la fonction de perte :

$$L_s^i = \sum_{(e_1, r, e_2) \in T_i^+ \cup T_i^-} log(1 + \exp(-y_{(e_1, r, e_2)} f_i(\mathbf{e_1}, \mathbf{r}, \mathbf{e_2})) \tag{5.4}$$

3.2.2 *Fonction objectif de distillation coopérative des connaissances*

Suivant l'objectif O2 (voir la section 3.1.2), la distillation des connaissances est menée de manière coopérative sur l'ensemble des n KBs. À chaque étape de l'apprentissage, chaque modèle de KB $\mathcal{M}^i$ participe à tour de rôle au processus élève enseignant. En tant qu'enseignant, le modèle distille son bagage de connaissances par le biais de la prédiction des classes L_s^i qui sont utilisées comme soft labels par les autres KB étudiantes pour calculer leur fonction de perte de mimétisme $L_{KD}^j, j = 1 \ldots n, j \neq i$. Mutuellement, en tant qu'étudiant, le modèle $\mathcal{M}^i$ utilise dans sa propre fonction objective de mimétisme L_{KD}^i les softs labels distillés par les autres KB enseignants à travers $L_s^j, j = 1 \ldots n, j \neq i$. Du point de vue de KB^i, la fonction de perte de distillation L_{KD}^i est formalisée comme la somme de deux pertes liées à la distillation de relations $L_{KD_r}^{ij}$ et à la distillation d'entités $L_{KD_e}^{ij}$ du réseau d'enseignants j au réseau d'étudiants i comme suit :

$$L_{KD}^i = \frac{1}{n-1} \sum_{j=1, j \neq i}^{n} L_{KD_r}^{ij} + L_{KD_e}^{ij} \tag{5.5}$$

Les fonctions de distillation quantifient la correspondance de la probabilité postérieure de chaque réseau d'étudiants avec celles fournies par les réseaux d'enseignants en ce qui concerne la plausibilité de l'estimation des embeddings donnée par les fonctions de classification supervisée respectivement f_i et f_j.

En suivant les principes de distillation des relations, la fonction objective de distillation $L_{KD_r}^{ij}$ quantifie la correspondance entre les distributions prédites par l'étu-

diant via la fonction f_i et les softs labels fournis par l'enseignant à partir de la fonction f_j. Les distributions de probabilités des relations utilisées par l'étudiant et l'enseignant sont obtenues à partir des scores de vraisemblance des triplets impliquant des relations alignées.

Les processus pour les entités sont similaire que pour les relations et suivent les principes de distillation des entités, la fonction objective de distillation $L^{ij}_{KD_e}$ quantifie la correspondance entre les distributions prédites par l'étudiant via la fonction f_i et les softs labels fournis par l'enseignant à partir de la fonction f_j. Les distributions de probabilités des entités utilisées par l'étudiant et l'enseignant sont obtenues à partir des scores de vraisemblance des triplets impliquant des entités alignées.

3.2.3 *Objectif de distillation des relations*

La distillation des relations entraîne le modèle étudiant $\mathcal{M}^i$ à imiter le modèle enseignant $\mathcal{M}^j$ via les probabilités prédites associées aux relations alignées $r \in I_r(i., j)$ tels que les triples (e^j_1, r, e^j_2) et (e^i_1, r, e^i_2) ont des scores de plausibilité proches. Ainsi, $L^{ij}_{KD_r}$ est calculé comme :

$$L^{ij}_{KD_r} = \sum_{(e^j_x, r, e^j_y) \in T^+_j \; : (e^i_x, e^j_x), (e^i_y, e^j_y) \in I_e(i,j)} \mathcal{D}(\mathcal{P}(r_{(e^j_x, \cdot, e^j_y)} \mid \theta^j), \mathcal{P}(r_{(e^i_x, \cdot, e^i_y)} \mid \theta^i)) \tag{5.6}$$

où $\mathcal{D}$ est la fonction de distillation qui peut être définie de plusieurs façons (Sau et Balasubramanian, 2016) telles que la distance L2 (Ba et Caruana, 2014) ou la divergence de Kullback-Leiber (Hinton *et al.*, 2015a), $r_{(e_x, \cdot, e_y)}$ est une variable catégorielle avec $|I_r(i,j)|$ dont les valeurs correspondent aux labels des relations alignées, $\mathcal{P}(r_{(e^j_x, \cdot, e^j_y)} \mid \theta^j)$ est une distribution catégorielle générée à partir des vrais triplets $(e^j_x, r, e^j_y) \in T^+{}_j$ et $\mathcal{P}(r_{(e^i_x, \cdot, e^i_y)} \mid \theta^i)$, une distribution catégorielle générée à partir des triplets impliquant les soft-labels associés aux relations fournies par le modèle $\mathcal{M}^j$.

Le score de confiance de la relation r_v est obtenu en convertissant les scores de plausibilité à l'aide de la fonction softmax sur les relations alignées $r \in I_r(i,j)$ comme ci-dessous :

$$\mathcal{P}^v(r_{(e_x, r_v, e_y)} \mid \theta^k) = \; softmax(f_k(\mathbf{e^k_x}, \mathbf{r_v}, \mathbf{e^k_x})) \tag{5.7}$$

où $k = i, j$ et,

$$softmax(f_k(\mathbf{e_x^k}, \mathbf{r_v}, \mathbf{e_x^k})) = \frac{\exp(f_k(\mathbf{e_x^k}, \mathbf{r_v}, \mathbf{e_x^k}))}{\sum_{r_w \in I_r(i,j)} \exp(f_k(\mathbf{e_x^k}, \mathbf{r_w}, \mathbf{e_x^k}))}, \tag{5.8}$$

la fonction softmax appliquée à chaque score de triplet.

3.2.4 *Objectif de distillation des entités*

La distillation des entités entraîne le modèle étudiant $\mathcal{M}^i$ à imiter le modèle enseignant $\mathcal{M}^j$ via les probabilités prédites associées aux entités alignées $e \in I_e(i,j)$ tels que les triples(e_x^j, r_w^j, e_y^j) et (e_1^i, r_v^i, e_2^i) ont des scores de plausibilité proches. Ainsi, $L_{KD_e}^{ij}$ est calculé comme :

$$L_{KD_e}^{ij} = \sum_{(e_x^j, r, e_y^j) \in T_j^+ \ : (e_x^i, e_x^j) \in I_e(i,j), (r_v^i, r_v^j) \in I_r(i,j)} \mathcal{D}(\mathcal{P}(e_{(e_x^j, r_{v,\cdot}^j)} \mid \theta^j), \mathcal{P}(e_{(e_x^i, r_{v,\cdot}^i)} \mid \theta^i)) \tag{5.9}$$

$r_{(e_x, r, \cdot)}$ est une variable catégorielle avec $|I_e(i,j)|$ valeurs correspondant aux labels des entités alignées, $\mathcal{P}(r_{(e_x,r,\cdot)} \mid \theta^j)$ est une distribution catégorielle générée à partir de l'ensemble des triplets $(e_x^j, r, e_y^j) \in T^+{}_j$ et $\mathcal{P}(r_{(e_x^j, r, \cdot)} \mid \theta^i)$, une distribution catégorielle générée à partir des triplets impliquant les soft-labels des relations fournis par le modèle $\mathcal{M}^j$.

Le score de confiance de l'entité e_y est obtenu en convertissant les scores de vraisemblance à l'aide de la fonction softmax appliquée sur les entités alignées $e \in I_e(i,j)$:

$$\mathcal{P}^y(r_{(e_x, r, e_y)} \mid \theta^k) = softmax(f_k(\mathbf{e_x}, \mathbf{r}, \mathbf{e_y})) \tag{5.10}$$

Avec la fonction softmax appliquée à chaque score des triplets

$$softmax(f_k(\mathbf{e_x}, \mathbf{r}, \mathbf{e_y})) = \frac{\exp(f_k(\mathbf{e_x}, \mathbf{r}, \mathbf{e_y}))}{\sum_{e_z \in I_e(i,j)} \exp(f_k(\mathbf{e_x}, \mathbf{r}, \mathbf{e_z}))}. \tag{5.11}$$

3.3 *Procédure d'entraînement du modèle KD-MKB*

Une caractéristique essentielle de la distillation coopérative des connaissances que nous proposons est que tous les objectifs $\mathcal{L}(\theta^i), i = 1 \ldots n$ des n modèles de

représentation des connaissances, sont optimisées conjointement et de manière coopérative. À chaque itération, chaque objectif $\mathcal{L}(\theta^i)$ utilise à la fois les labels réels et les softs labels fournies par les modèles $\mathcal{M}^j, j = 1 \dots n, j \neq i$ pour mettre à jour les paramètres θ^i. Le modèle d'apprentissage est résumé dans l'algorithme 2.

La stratégie d'apprentissage est configurée dans chaque mise à jour du modèle basée sur un mini-batch. À chaque itération, tous les objectifs $\mathcal{L}(\theta^i)$ sont appris conjointement en utilisant un mini-batch dans le cadre de l'apprentissage de L_s^i et $(n-1)$ mini-batchs comprenant des paires d'alignements dans le cadre de l'apprentissage de $L_{KD_e}^{ij}$ et $L_{KD_r}^{ij}$. Étant donné que les tailles des ensembles d'entités et de relations utilisées dans le calcul de normalisation softmax de $\mathcal{P}(.)$ de $\mathcal{E}^i$ ou $\mathcal{R}^i$ peuvent être très grands, nous appliquons une technique d'échantillonnage pour estimer la distribution de probabilité comme cela a été fait dans les travaux précédents (Liu *et al.*, 2018). Elle consiste, *pour les entités*, à sélectionner les k top entités candidates du triplet à distiller plus k entités aléatoires. Le modèle qui joue le rôle de l'enseignant est chargé de choisir les meilleures entités. Par conséquent, la fonction softmax n'utilise que $2 \times k$ entités pour la normalisation au lieu de $|\mathcal{E}^i|$ ou $|I_e(i,j)|$ valeurs totales ce qui réduit drastiquement le nombre de calculs requis pour chaque mini-batch de distillation. D'une manière similaire, la technique d'échantillonnage, *pour les relations*, consiste à sélectionner les k top relations candidates du triplet à distiller plus k relations aléatoires. Le modèle qui joue le rôle de l'enseignant est chargé de choisir les meilleures relations. Par conséquent, la fonction softmax n'utilise que $2 \times k$ relations pour la normalisation au lieu de $|\mathcal{R}^i|$ ou $|I_r(i,j)|$ valeurs totales ce qui réduit, comme pour les entités, le nombre de calculs requis pour chaque mini-batch de distillation.

Algorithme 2 : Procédure d'entraînement de KD-MKB

Input : KD-MKB paramètre α, Modèle de représentation de KB avec ses propres paramètres Ensemble de bases de connaissances $\mathcal{KB} = \{KB^1, KB^2, \ldots, KB^n\}$ Ensemble d'entités et de relations alignées $I_e(i,j)$, $I_r(i,j)$ $\forall i,j \in \{1,..,n\}$ with $i \neq j$

Initialisation : Ensemble de modèles $\mathcal{M} = \{\mathcal{M}^1, \mathcal{M}^2, \ldots, \mathcal{M}^n\}$

tant que *convergence ou nombre maximal d'itérations non atteint* **faire**

  pour $\mathcal{M}^i \in \mathcal{M}$ **faire**

    index_top-k $\leftarrow$ création de l'index avec M^i

    $batch^i \leftarrow$ échantillonne les triplets deT$_i^+$ et T_i^-

    $\mathcal{L}^i \leftarrow$(1-α) $\times$ $\mathcal{L}^i$ avec $batch^i$ suivant eq. (5.4)

  pour $\mathcal{M}^i \in \mathcal{M}$ **faire**

    pour $\mathcal{M}^j \in \mathcal{M}$ **faire**

      si $i \neq j$ **alors**

        $batch_e^j \leftarrow$ requête index top-k avec batch$^i \cap I_e(i,j)$ pour obtenir $\mathcal{P}(g_{(e_x^i, e_y^i)}|\theta^i)$

        $batch_r^j \leftarrow$ requête index top-k avec batch$^i \cap I_r(i,j)$ pour obtenir $\mathcal{P}(g_{(e_x^i, r_v^i)}|\theta^i)$

        $L_{KD}^j \leftarrow$ distille professeur M^i vers l'étudiant $\mathcal{M}^j$ avec eq. (5.5) et $batch_r^j$, $batch_e^j$

        $\mathcal{L}^j \leftarrow \mathcal{L}^j + \alpha \times L_{KD}^j$

  pour $\mathcal{M}^i \in \mathcal{M}$ **faire**

    Mise à jour conjointe $\mathcal{M}^i$ w.r.t. $\mathcal{L}^i$

Sortie : Paramètres θ^i pour chaque modèle $\mathcal{M}^i$

4 Cadre expérimental

Deux objectifs principaux ont guidé nos expériences : 1) montrer la validité de la distillation des connaissances pour soutenir formellement le transfert de connaissances entre KBs ; 2) évaluer l'efficacité du modèle KD-MKB.

4.1 *Configuration*

4.1.1 *Configuration des graphes de connaissances*

Nous réalisons nos expériences sur deux KB standard, WN18RR et FB15K-237 KBs [1], décrites dans la Section 2.

Nous simulons le cadre des KBs multiples en divisant aléatoirement chacun des triples d'entraînement des KBs WN18RR et FB15K-237 en 2 et 3 partitions ($n = 2, 3$ dans le cadre de KD-MKB). Notre motivation pour ce cadre d'évaluation est soutenue par deux raisons :

1. notre objectif avec *KD-MKB* est d'apprendre des embeddings de KB autonomes au lieu d'embeddings multigraphes
2. évaluer l'effet intrinsèque du modèle *KD-MKB* sans biais induit par un effet non contrôlé de la qualité de l'alignement des connaissances. Plus précisément, deux partitions FB15K-237 partagent généralement 95% des entités mais l'alignement chute à 64% pour WN18RR. Le paramètre $n = 1$ permet de rapporter les résultats des modèles traditionnel sur l'ensemble des triplets.

Le tableau 1 fournit des statistiques pour les deux ensembles de données utilisés dans nos expériences. Dans chaque cas, nous obtenons n modèles de l'enseignant $\mathcal{M}^i$ et de l'étudiant $\mathcal{M}^j$, les résultats présentés sont donc des moyennes. Nous comparons les performances de notre modèle à celles des modèles de représentation les plus récents, à savoir TransE (Bordes *et al.*, 2013). Nous nous concentrons sur la tâche standard de prédiction de liens entre entités pour la population de bases de connaissances. Cette tâche évalue les performances du modèle pour une requête visant à compléter le triplet (e_i,r_j,e ?) où la réponse est une liste classée d'entités qui correspondent mieux à e ? (de la même manière, les requêtes de concernant la tête des triplets peuvent être évaluées). Nous utilisons les métriques standards HITS@k ($k = 1, 3, 10$) et MRR. Nous rapportons les moyennes de plusieurs exécutions sur des partitions de test.

1. Les deux KBs sont disponibles sur https ://www.microsoft.com/en-us/download/details.aspx ?id=52312 (Dernière vérification le 18/08/2022)

4.1.2 *Configuration des stratégies de distillation des connaissances*

Nous analysons l'efficacité des stratégies de distillation des connaissances en comparant les résultats rapportés par les scénarios suivants :

1. *Independant* est la configuration traditionnelle de distillation à sens unique (Hinton *et al.*, 2015a) où seulement la moitié ($n = 2$) ou le tiers ($n = 3$) de la connaissance est transférée de l'enseignant à l'étudiant respectif. Le modèle de l'enseignant est pré-entraîné et fournit des prédictions a posteriori d'entités et de relations au modèle de l'étudiant. Notez que dans ce cadre, chaque modèle KB joue de façon statique le rôle de l'enseignant ou de l'étudiant pendant le processus d'aprentissage
2. *Xdistills*$\sim X$ est une configuration séquentielle dans laquelle chaque modèle est d'abord entraîné sur l'un des ensembles définis dans la partition jusqu'à convergence. Ensuite, il joue le rôle d'un professeur et distille ses connaissances à d'autres modèles qui jouent le rôle d'étudiants, puis, il joue le rôle d'étudiant. Ainsi, les paramètres des modèles KB enseignant et KB étudiant sont mis à jour l'un après l'autre de manière séquentielle. Cependant, l'utilisation d'une distillation séquentielle garantit que le modèle voit les triplets d'une seule partition. Le principal inconvénient est que certaines des connaissances apprises en tant que professeur peuvent être perdues lorsqu'on se comporte comme un étudiant.
3. *KD-MKB* procédure dans laquelle chaque modèle agit dynamiquement et simultanément comme enseignant et comme élève pendant tout le processus d'entraînement. Ainsi, les prédictions et les paramètres des modèles KB sont mis à jour conjointement.

4.1.3 *Détails de l'implémentation*

Nous avons implémenté notre modèle en utilisant PyTorch [2]. La fonction de perte est minimisée en utilisant l'optimiseur Adam avec un taux d'apprentissage de 10^{-5}. Le nombre maximal d'itérations est fixé à 8×10^4. Les paramètres du modèle TransE ont été fixés en sélectionnant la meilleure configuration dans la partition de validation de chaque jeu de données et en suivant les recommandations de (Sun *et al.*, 2019b). Ainsi, la taille des représentations est fixée à 1000 (resp. 500), la taille du lot à 512 (resp. 256), la taille d'échantillonnage négatif à 128 (resp. 512), la perte adversariale α^{al} à 1 (resp. 0,5), et l'hyperparamètre de marge γ à 9 (resp. 6) pour FB15K-237 (resp. WN18RR). Les entités du top-k sont trouvées en utilisant la librairie faiss (Johnson *et al.*, 2017) avec k fixé à 10. L'hyperparamètre

2. Notre implémentation est disponible publiquement à l'adresse https ://github.com/raphaelsty/mkb

α dans l'équation 5.1, est fixé à 0,98 suite à l'analyse des mesures des fonctions objectives L_s^i et L_{KD}^i.

4.2 *Analyse du modèle KD-MKB*

4.2.1 *La distillation des connaissances entre KBs fonctionne-t-elle ?*

	WN18RR							
	Professeur				Étudiant			
n	HITS@1	HITS@3	HITS@10	MRR	HITS@1	HITS@3	HITS@10	MRR
1	1.43	39.61	52.63	0.22	1.37	39.53	52.40	0.22
2	0.65	19.90	28.36	0.11	0.62	19.75	28.35	0.11
3	0.58	12.12	18.61	0.07	0.59	11.84	18.10	0.07

TABLEAU 5.1 – Résultats pour les ensembles de données WN18RR sur les tâches de prédiction des liens en utilisant le modèle traditionnel de distillation indépendant. n indique le nombre de partitions de KB utilisées pour apprendre la représentation de l'enseignant. Pour $n > 1$, les valeurs indiquées correspondent aux performances moyennes des multiples modèles sur l'ensemble de test.

	FB15K-237							
	Professeur				Étudiant			
n	HITS@1	HITS@3	HITS@10	MRR	HITS@1	HITS@3	HITS@10	MRR
1	22.53	36.27	52.15	0.32	22.46	36.33	52.24	0.32
2	19.28	31.56	46.23	0.28	19.18	31.35	45.99	0.28
3	17.19	28.35	42.59	0.26	17.09	28.18	42.11	0.25

TABLEAU 5.2 – Résultats pour les ensembles de données FB15K-237 sur les tâches de prédiction des liens en utilisant le modèle traditionnel de distillation indépendant. n indique le nombre de partitions de KB utilisées pour apprendre la représentation de l'enseignant. Pour $n > 1$, les valeurs indiquées correspondent aux performances moyennes des multiples modèles sur l'ensemble de test.

A notre connaissance, ce travail est la première tentative dans la littérature d'évaluer empiriquement l'inférence de connaissances entre les représentations de KBs en utilisant le cadre théorique de la distillation de connaissances (Hinton *et al.*, 2015a).

Les tableaux 5.1 et 5.2 montrent les performances de prédiction des liens pour les deux ensembles de données utilisés lors de la procédure de distillation indépendante d'un enseignant $\mathcal{M}^i$ à un étudiant $\mathcal{M}^j$. Nous pouvons voir dans ces tableaux que, dans l'ensemble, les niveaux de performance du modèle de l'élève

suivent ceux du modèle de l'enseignant pour toutes les mesures et que les tendances de performance restent les mêmes pour un nombre croissant de KBs. Ce résultat valide empiriquement notre idée sur la modélisation de l'inférence de connaissances entre KBs par la formalisation de la distillation simultanée des entités ($L^{ij}_{KD_e}$) et des relations ($L^{ij}_{KD_r}$).

Distillation strategy	n	HITS@1			HITS@3			HITS@10			MRR		
		Best	Worst	%Chg.	Best	Worst	%Chg	Best	Worst	%Chg	Best	Worst	%Chg
Indépendant	2	0.70	0.60	+455.7%	19.95	19.86	+68.6%	28.38	28.35	+48.6%	0.11	0.11	+72.7%
	3	0.74	0.28	+173.0%	12.53	11.59	+68.4%	19.78	17.32	+70.5%	0.07	0.07	+85.7%
Xdistills∼X	2	1.21	1.03	+221.5%	27.93	27.69	+20.4%	41.75	41.48	+1.0%	0.16	0.15	+18.8%
	3	1.57	1.26	+28.7%	16.51	16.19	+27.8%	30.47	30.42	+11.0%	0.10	0.10	+30.0%
KD-MKB	2	3.89	3.74	-	33.64	33.55	-	42.18	42.02	-	0.19	0.19	-
	3	2.02	1.75	-	21.10	19.57	-	33.72	32.72	-	0.13	0.12	-

Tableau 5.3 – Résultats des stratégies de distillation pour le jeu de données WN18RR sur la tâche de prédiction des liens. Les valeurs rapportées sont les meilleures et les pires performances obtenues dans chaque configuration de fractionnement de n dataset. %Chg. indique l'amélioration de l'efficacité du modèle KD-MKB par rapport aux stratégies de distillation concurrentes envisagées sur la base du modèle le plus performant.

Distillation strategy	n	HITS@1			HITS@3			HITS@10			MRR		
		Best	Worst	%Chg.	Best	Worst	%Chg	Best	Worst	%Chg	Best	Worst	%Chg
Indépendant	2	19.38	19.18	+24.8%	31.92	31.21	+18.3%	46.45	46.02	+14.7%	0.28	0.28	+17.9%
	3	17.32	17.03	+41.1%	28.55	28.22	+32.7%	42.88	42.43	+23.7%	0.25	0.25	+36.0%
Xdistills∼X	2	20.43	20.39	+18.4%	32.60	32.04	+15.8%	48.15	47.85	+10.6%	0.29	0.29	+13.8%
	3	18.85	18.70	+29.7%	29.61	29.50	+27.9%	44.73	44.46	+18.6%	0.27	0.27	+25.9%
KD-MKB	2	24.18	24.12	-	37.75	37.66	-	53.26	53.22	-	0.33	0.33	-
	3	24.44	24.36	-	37.88	37.82	-	53.06	52.95	-	0.34	0.33	-

Tableau 5.4 – Résultats des stratégies de distillation pour le jeu de données FB15K-237 sur la tâche de prédiction des liens. Les valeurs rapportées sont les meilleures et les pires performances obtenues dans chaque configuration de fractionnement de n dataset. %Chg. indique l'amélioration de l'efficacité du modèle KD-MKB par rapport aux stratégies de distillation concurrentes envisagées sur la base du modèle le plus performant.

4.2.2 *Modèle de distillation*

Pour mettre en évidence les avantages de la distillation coopérative à travers les bases de connaissances, nous présentons dans le tableau 5.4 les résultats de la tâche de prédiction des liens en utilisant trois stratégies de distillation des connaissances : *Indépendant*, *Xdistills*$\sim X$ et *KD-MKB* en utilisant les mêmes partitions de données que celles présentées dans le tableau **??**. Nous présentons les meilleurs et les pires résultats des modèles $\mathcal{M}^i$ et $\mathcal{M}^j$ pour chaque partition des KBs.

La principale observation que l'on peut tirer du tableau 5.4 est que le modèle *KD-MKB* surpasse le modèle *Indépendant* (par exemple, entre 24,8% et 455,7% d'amélioration sur la base de HITS@1, entre 17,9% et 85,7% d'amélioration sur la base de MRR) sur toutes les partitions et tous les ensembles de données et pour toutes les métriques. En outre, nous pouvons également constater que le modèle *Xdistills*$\sim X$ est plus performant que le modèle *Indépendant*. Par exemple, lorsque $n = 2$, la performance de HITS@3 atteint un niveau d'environ 27,93 (resp. 32,60) avec le modèle *Xdistills*$\sim X$ pour le jeu de données WN18RR (resp. FB15K-237) contre une valeur plus faible d'environ 19,95 (resp. 31,92) avec le modèle *Indépendant*. Cela s'explique facilement par le fait que le modèle *Indépendant* n'est entraîné que sur les softs labels d'une seule partition, tandis que le modèle *Xdistills*$\sim X$ utilise d'abord les labels de sa propre partition (lorsqu'il joue le rôle d'un enseignant), puis les soft-labels des autres partitions (lorsqu'il joue le rôle d'un étudiant). Il est également intéressant de noter que le modèle *KD-MKB* est plus performant que le modèle *Xdistills*$\sim X$, même si le pourcentage de progression est plus faible (par exemple, entre 1,0 et 11,0% d'amélioration pour HITS@10 avec la base de données WN18RR, entre 10,6 et 18,6% d'amélioration pour HITS@10 avec la base de données FB15K-237). Il convient de mentionner que le modèle *KD-MKB* utilise le même nombre de labels que le modèle *Xdistills*$\sim X$. Ce résultat met en évidence un avantage clair des changements dynamique entre les rôles d'enseignant et d'étudiant pour chacun des modèles $\mathcal{M}^i$, renforcée par l'apprentissage coopératif par mimétisme et la mise à jour conjointe de leurs paramètres.

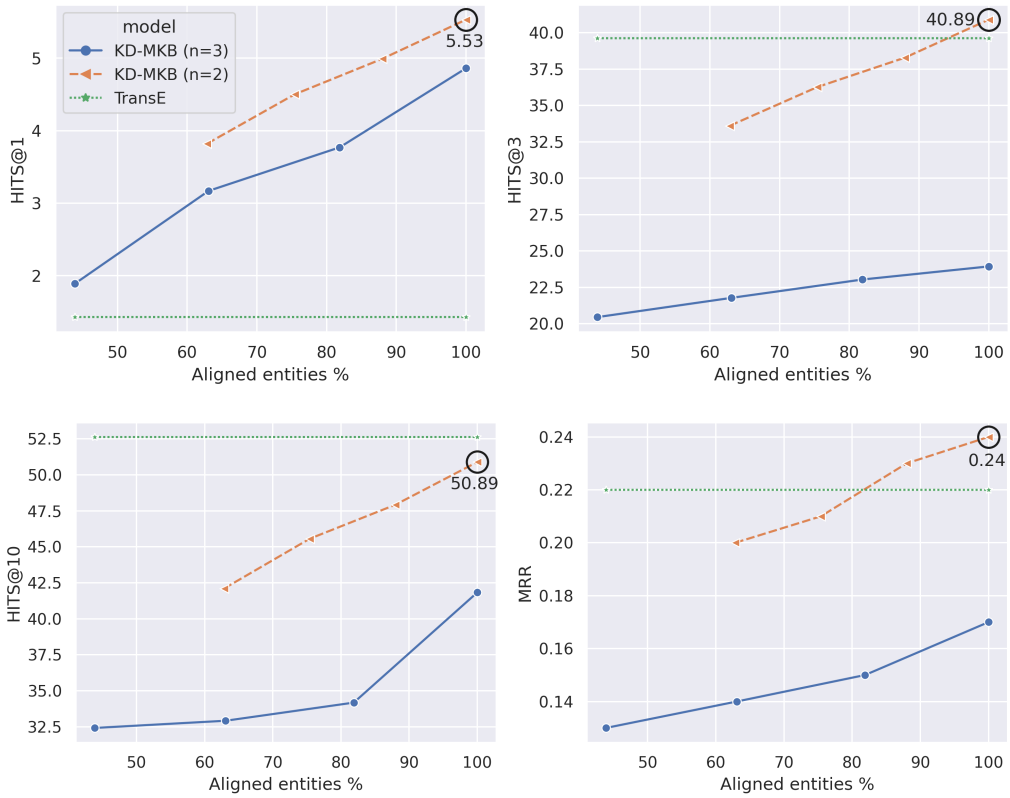


FIGURE 5.3 – Résultats de prédiction de liens HITS@1, HITS@3, HITS@10 et MRR pour *KD-MKB* en utilisant WN18RR lorsque différentes tailles de l'ensemble d'alignement ($I_e(i,j)$) sont utilisées. Nos meilleures performances sont mises en évidence par un cercle et les valeurs ont été incluses.

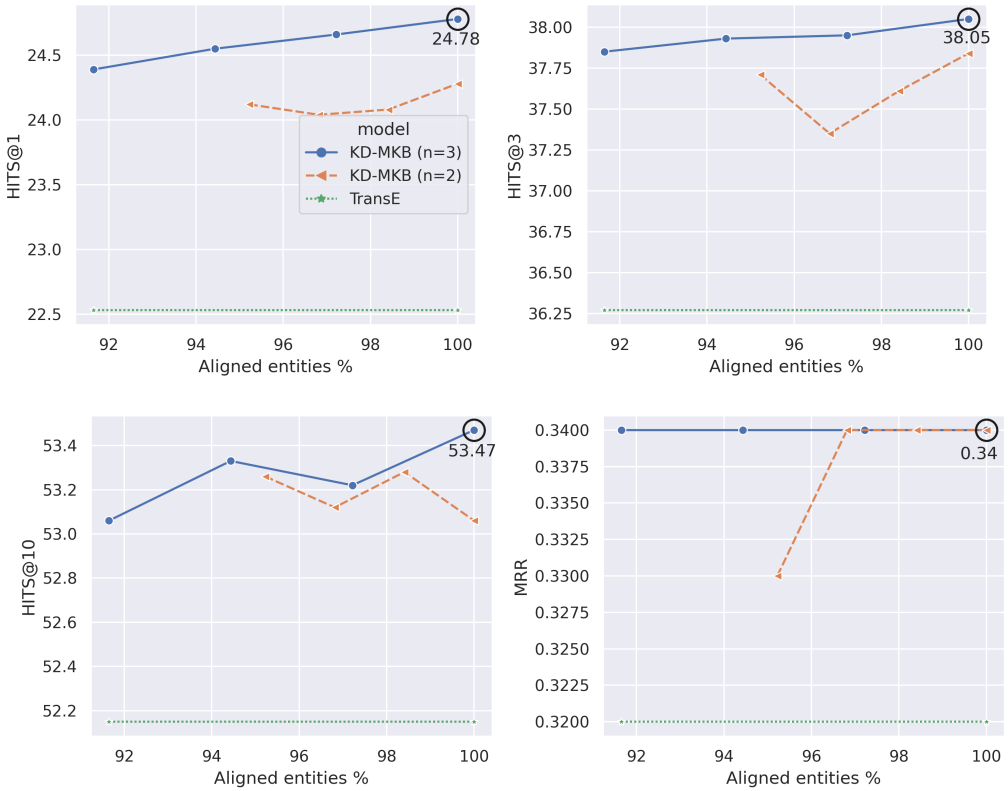


FIGURE 5.4 – Résultats de prédiction de liens HITS@1, HITS@3, HITS@10 et MRR pour *KD-MKB* en utilisant FB15K-237 lorsque différentes tailles de l'ensemble d'alignement ($I_e(i,j)$) sont utilisées. Nos meilleures performances sont mises en évidence par un cercle et les valeurs ont été incluses.

4.2.3 *Apprentissage multi-KB par rapport à l'apprentissage mono-KB.*

Nous analysons également l'effet de la taille des alignements d'entités sur les performances de *KD-MKB*. Pour être cohérent avec les résultats présentés dans les expériences précédentes, nous conservons les mêmes partitions. La figure 5.3 représente les variations de performance par rapport à toutes les métriques en utilisant le jeu de données WN18RR. Il convient de mentionner que, contrairement au FB15K-237 qui présente un chevauchement de 95%, le WN18RR permet en fait de simuler un chevauchement croissant des entités en ajoutant des informations supplémentaires à $I_e(i,j)$ jusqu'à atteindre un chevauchement de 100%. L'alignement de d'entités supplémentaire n'augmente pas le nombre de triplets à entraîner, mais permet de distiller un plus grand nombre d'entités à partir des enseignants. La figure 5.3 indique qu'en moyenne, 20% d'entités alignées supplémentaires entraînent un gain absolu de $5,2$ points en HITS@10 et de $0,06$ points

en MRR. Comme prévu, un nombre plus élevé de soft labels améliore l'inférence de connaissance mutuelle d'une base de données à ses pairs. Les résultats pour les ensembles de données FB15K-237 sont présentés dans la figure 5.4. Dans ce cas, l'augmentation est bénéfique, étant plus stable lorsque $n = 3$ en raison de l'augmentation plus importante des entités superposées.

	WN18RR				FB15K237			
Model	HITS@1	HITS@3	HITS@10	MRR	HITS@1	HITS@3	HITS@10	MRR
TransE	1.43	39.61	**52.63**	0.22	22.53	36.27	52.15	0.32
KD-MKB	**5.64**	**41.49**	51.70	**0.24**	**24.28**	**37.84**	**53.06**	**0.34**

TABLEAU 5.5 – Résultats pour les ensembles de données WN18RR et FB15K-237 sur la tâche de prédiction de liens à l'aide des modèles TransE et KD-MKB partageant 100% d'entités et de relations. Pour KD-MKB, les valeurs indiquées correspondent aux performances moyennes des différents modèles sur les ensembles de tests.

4.2.4 *Distillation avec des alignements plus importants.*

Les figures 5.3 et 5.4 illustrent les performances du modèle *KD-MKB* ainsi que celles du modèle TransE en utilisant les deux ensembles de données FB15k-237 et WN18RR. Les deux modèles ont été entraînés à l'aide de l'ensemble des données d'entraînement. TransE est entraîné à partir de l'ensemble des triplets. *KD-MKB* est entraîné à partir des triplets d'entraînement et des softs labels. Les résultats sont constants pour TransE car ses performances ne dépendent pas du nombre d'entités alignées. Lorsque le nombre d'entités alignées est de 100% (mis en évidence par un cercle), nous pouvons vérifier à partir de ces figures les améliorations apportées par l'utilisation de *KD-MKB* sur plusieurs KBs au lieu du modèle de représentation classique TransE sur un KB individuel. Pour FB15K-237, les deux configurations de partitions (par exemple *KD-MKB* avec $n = 2$ et $n = 3$) surpassent les performances de TransE pour toutes les métriques étudiées. Cependant, pour WN18RR, *KD-MKB* surpasse légèrement TransE lorsque $n = 2$ en termes de HITS@3 et MRR, mais ne parvient pas à améliorer en termes de HITS@10.

5 Bilan

Cette contribution présente un nouveau cadre pour l'apprentissage des représentations d'entités et de relations sur plusieurs bases de connaissances. Le cadre que nous avons défini exploite une nouvelle façon de transférer l'apprentissage d'un modèle de KB à ses pairs. Tout d'abord, nous formalisons l'inférence des entités et des relations entre les bases de connaissance comme un objectif de distillation sur les distributions de probabilité postérieures via les connaissances alignées. Sur la base de cette découverte, nous proposons et formalisons un cadre de distillation coopératif dans lequel un ensemble de modèles de KB sont appris conjointement en utilisant chacun les étiquettes associées à la vérité terrain provenant de leur propre contexte et également des *soft labels* fournies par des pairs. Nous démontrons empiriquement le raisonnement qui sous-tend la distillation des connaissances entre les KB et montrons l'efficacité de notre cadre d'apprentissage coopératif sur la tâche de prédiction des liens par rapport aux stratégies de distillation existantes.

Pour résumer, dans ce chapitre un cadre basé sur la distillation a été proposé et analysé. Cependant, il est limité à l'interaction entre bases de connaissances. Dans le prochain chapitre, nous étendrons le cadre présenté dans ce chapitre pour faire intervenir les modèles des langues pré-entraînés en tant que professeur et en tant qu'étudiant.

6

ENRICHISSEMENT DES MODÈLES DE LANGUE PRÉ-ENTRAÎNÉS PAR LA DISTILLATION MUTUELLE DES CONNAISSANCES

1 Contexte et motivations

Les systèmes de complétion de connaissances, dont le but est la construction ou la mise à jour continue des bases de connaissances (KB), s'appuient de plus en plus ces dernières années sur des modèles de langue pré-entraînés (PLM). Les modèles de l'état de l'art qui résolvent cette tâche sont généralement basés sur un cadre de recherche et de lecture (*retriever-reader*), dans lequel le système filtre d'abord les documents candidats dans un cadre de recherche et réordonnancement, puis extrait ou génère les réponses, en se basant sur les informations identifiées dans les documents filtrés. Les PLMs permettent d'atteindre dans cette chaîne, un niveau de performance élevé tant en qualité de système lecteur (*reader*) que système de ré-ordonnancement (*retriever*).

Dans cette contribution, nous formulons une procédure originale de pré-entraînement coopératif et améliorons le PLM pour traiter les entités pour une tâche orientée connaissances. Nous montrons que l'intégration d'informations riches et annotées manuellement sur les entités des KBs dans les modèles de représentation du langage conduit à une amélioration globale des candidats. Nous explorons une nouvelle direction en entraînant mutuellement les PLMs et les modèles de représentationdes KBs via la distillation des connaissances. Nous soutenons que le corpus textuel et la base de connaissance représentent deux espaces distincts et complémentaires pour les tâches nécessitant des connaissances spécifiques. L'entraînement de ces modèles via un environnement dynamique, où les rôles entre les enseignants et les étudiants s'échangent, permet d'améliorer la capacité de traitement des entités et d'obtenir des résultats compétitifs sur deux ensembles de données standards dédiés à des tâches orientées connaissances, comme le slot filling, à savoir T-REx et zsRE.

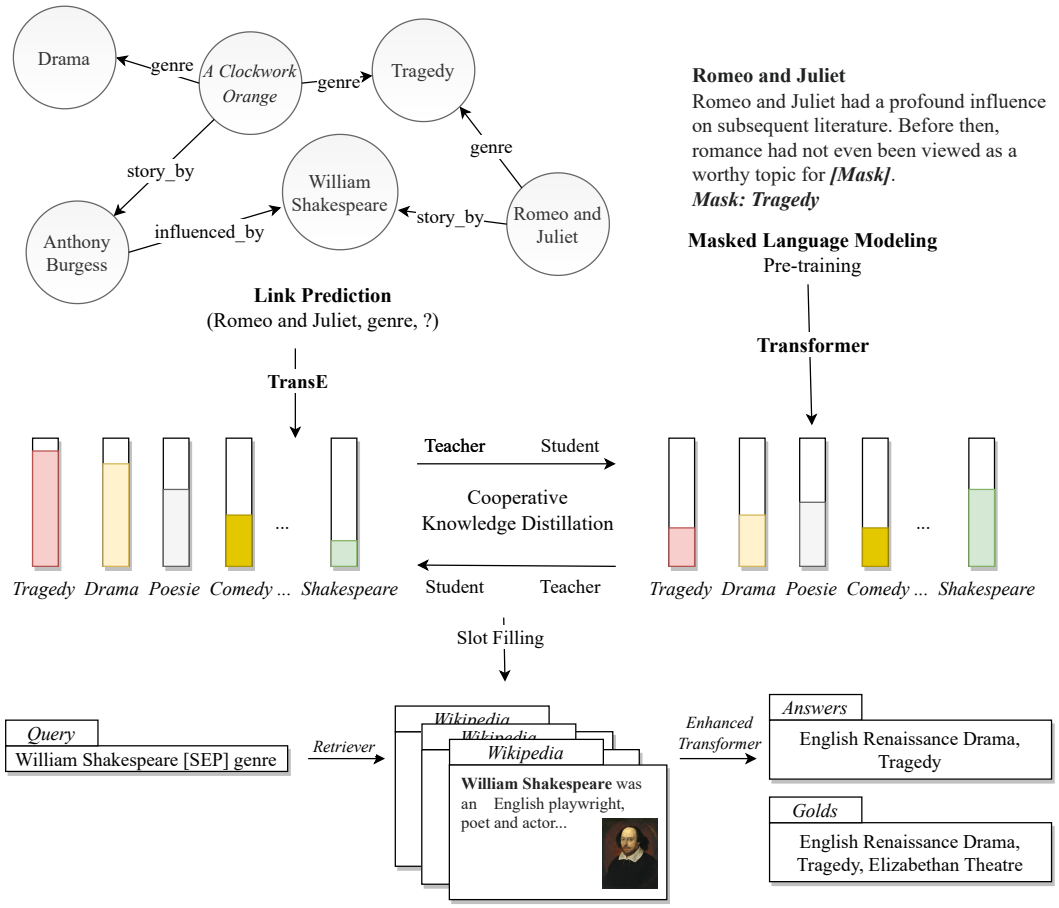


FIGURE 6.1 – Distillation coopérative des connaissances entre le PLM et la KB pour la tâche de slot filling.

Les bases de connaissances contiennent des informations structurées sur les entités (nœuds) et les relations (arêtes) sous la forme de triplets. Par exemple, le triplet *[William Shakespeare, genre, tragédie]* utilise la relation *genre* pour fournir des informations sur l'entité *William Shakespeare*. Ces informations sont aujourd'hui de plus en plus exploitées comme une source structurée qui profite à diverses tâches de TAL à forte intensité de connaissances. Cependant, les bases de données ne sont pas toujours complètes. La découverte de nouveaux faits sur les entités à partir d'un corpus textuel et l'ajout de ces faits à la KB est une tâche difficile communément appelée population de base de connaissances (KBP) (Surdeanu et Ji, 2014; Balog, 2018). Dans la littérature, les pipelines de KBP comprennent généralement une résolution de coréférence d'entité, une reconnaissance/liaison d'entité, et une méthode d'extraction/classification de relation (Ellis *et al.*, 2015). Cependant, les modèles traditionnels de classification supervisée des relations suivent

l'hypothèse d'un monde fermé et ignorent les relations nouvelles. Ils souffrent donc d'un manque de capacité de généralisation (Ren *et al.*, 2020).

Pour surmonter ces difficultés, des travaux récents se sont concentrés sur les approches de retriever-reader avec des modèles de langage (PLM) pré-entraînés (Lewis *et al.*, 2020b; Guu *et al.*, 2020) et suivant des hypothèses de monde ouvert. Parmi d'autres techniques, le slot filling est une stratégie utilisée à cette fin. Les systèmes de slot filling récupèrent les entités candidates susceptibles de combler les informations manquantes au sein d'un triplet *[entité, créneau, ?]* en s'appuyant sur des preuves solides provenant d'un large corpus. Un exemple récent dans cette direction de recherche est le classement KILT (Petroni *et al.*, 2021) qui a popularisé deux datasets de slot filling, T-REx (Elsahar *et al.*, 2018) et zsRE (Levy *et al.*, 2017). KILT fournit un sous-ensemble de Wikipedia comme source de connaissances externe.

Des recherches récentes sur les PLMs augmentés ont attiré l'attention et démontré que diverses tâches extrinsèques bénéficient de sources de connaissances structurées et annotées manuellement (Yamada *et al.*, 2020; Zhang *et al.*, 2019d; Wang *et al.*, 2021b; Bevilacqua et Navigli, 2020). Cet article étudie la capacité des PLMs enrichis à partir des informations des KBs à améliorer la génération de candidats sous l'objectif de slot filling. Nous proposons un cadre original de distillation coopératif des connaissances pour aligner la tâche de pré-entraînement de la modélisation du langage masqué (MLM) des PLMs (Devlin *et al.*, 2019) et l'objectif de prédiction de liens des modèles de représentation des KBs (Bordes *et al.*, 2013; Yang *et al.*, 2015b; Sun *et al.*, 2019b). L'idée clé de notre travail est de transférer de manière coopérative les connaissances d'une KB et d'un PLM grâce à l'utilisation de soft labels et à la régularisation via la distillation des connaissances. Par exemple, *{Drama, Tragedy, .., Comedy}* sont des entités du jeu de données FB15K-237 (Toutanova *et al.*, 2015) qui compléteraient probablement la phrase masquée *"Le genre de Roméo et Juliette est [Mask]"*. Optimiser le PLM pour récupérer à la fois la vérité terrain et retrouver les entités les plus similaires à la cible permet de l'entraîner à répondre *"drama"*.

Un exemple de notre hypothèse est représenté dans la figure 6.1. Les PLMs retrouvent un mot masqué tel que *'tragedy'* pour la phrase *"Romeo and Juliet is a [Mask]"* pendant la phase de pré-entraînement. Nous proposons plutôt d'entraîner notre PLM à récupérer non seulement le mot masqué associé à la vérité terrain, mais à produire des logits plus élevés pour les entités qui sont susceptibles de remplacer la vérité terrain, c'est-à-dire, les mots *{"Drama", "Tragedy", "Poetry", .., "Comedy"}*, où ces soft-labels de haute qualité sont obtenues à partir d'un modèle de représentation de KB, comme en utilisant la tâche de prédiction de lien avec la fonction de score TransE (Sourty *et al.*, 2020). La figure 6.1 présente un petit sous-ensemble du graphe de connaissances FB15K-237, où un modèle TransE entraîné

est utilisé pour retrouver les principales entités susceptibles de compléter le triplet *[Roméo et Juliette, genre, ?]*.

Nous montrons qu'un tel objectif de pré-entraînement est complémentaire à l'objectif de slot filling et améliore la qualité des candidats récupérés par le PLM. Au lieu d'une connexion unique entre la KB et le PLM, nous proposons d'apprendre les représentations de la KB et les paramètres du PLM en suivant un cadre coopératif pour améliorer les modèles mutuellement. Les deux modèles jouent successivement le rôle de professeur et d'élève pour transmettre des softs labels et agissent comme un régularisateur l'un pour l'autre. Par conséquent, nos principales contributions sont les suivantes :

— Une stratégie coopérative de distillation des connaissances axée sur les entités entre le MLM (PLM) et la procédure de prédiction de liens (modèle de plongement de la KB).
— Une évaluation approfondie des PLM standard par rapport à nos versions de PLM enrichis sur des tâches orientées connaissances utilisant deux ensembles de données standards, T-REx et zsRE.

2 Problématiques et définitions

2.1 *Concepts et définitions*

Considérons un KB comme un graphe $(\mathcal{E}, \mathcal{R})$ composé d'entités $\mathcal{E} = \{e_1, \ldots, e_{N_e}\}$, d'un ensemble de relations $\mathcal{R} = \{r_1, \ldots, r_{N_r}\}$, et un ensemble de triplets positifs, ou faits, (e_x, r_w, e_y) noté T^+ parmi tous ceux possibles dans $\mathcal{E} \times \mathcal{R} \times \mathcal{E}$.

D'un point de vue formel, la tâche de complétion de slot peut être définie comme suit : étant donné une requête composée d'une entité $e_i \in \mathcal{E}$ et d'un slot de type $s_k \in \mathcal{R}$, l'objectif de la tâche consiste à retrouver une entité $e_j \in \mathcal{E}$ pour reconstruire le triplet positif (e_i, s_k, e_j). Les requêtes (e_i, s_k) peuvent conduire à des réponses cibles multiples $\{e_j^0, e_j^1, .., e_j^n\}$ où les entités e_k^i sont soit des mentions distinctes de la même entité, soit deux entités différentes. Notez que la reconstruction est possible pour tout modèle capable d'utiliser (e_i, s_k) comme entrée et de produire un ensemble d'entités candidates.

2.2 *Définition du problème*

L'objectif de notre modèle est double : 1) améliorer la compréhension de la requête par le modèle en enseignant des représentations de l'ensemble d'entités de $\{e_i^0, e_i^1, .., e_i^n\}$ qui sont susceptibles de remplacer l'entité e_i sur la base de son voisinage dans l'espace des représentations KB ; 2) augmenter la capacité du modèle à proposer des candidats pertinents en apprenant un ensemble de représentations d'entités qui peuvent être utilisées comme substituts de l'ensemble des réponses attendues $\{\{e_{j_0}^0, e_{j_1}^0, .., e_{j_m}^0\}, .., \{e_{j_0}^n, e_{j_1}^n, .., e_{j_m}^n\}\}$ où $\{e_{j_0}^0, e_{j_1}^0, .., e_{j_m}^0\}$ est l'ensemble des entités qui peuvent remplacer la réponse attendue e_j^0.

Les modèles d'apprentissage des représentations de KB, tels que TransE Bordes *et al.* (2013), apprennent des représentations des entités tout en prenant en compte sa structure particulière en tant qu'ensemble d'entités connectées par des relations. Les connaissances distillées par TransE permettent aux PLM de comprendre les similarités / différences entre les entités et agissent comme un régularisateur. Réciproquement, le modèle de représentation de KB tire parti des connaissances acquises par le PLM à mesure de son entraînement. Nous soutenons que la mise à jour simultanée des modèles de KB embeddings et des PLMs permet de construire un espace où la distillation des connaissances est plus facile à opérer.

3 PLM enrichi via la distillation coopérative des connaissances

3.1 *Procédure d'augmentation du modèle de langue*

Dans cette partie, nous détaillons la procédure d'augmentation du modèle de langue et les moyens que nous avons mis en place pour aligner les tâches de MLM (PLM) et de prédiction de liens (KB).

3.1.1 *Alignement des probabilités des entités*

Nous avons aligné les tâches de prédiction des liens et de MLM pour transférer les connaissances encodées par le modèle de représentation des KB. Lors de la tâche MLM, nous masquons 30% du temps une entité afin d'entraîner notre modèle via notre objectif KD (voir la section 3.2), et 70% du temps, nous appliquons la procédure standard définie par Devlin *et al.* (2019). Le modèle doit retrouver le mot original avec une entropie croisée lorsqu'il s'agit de la procédure standard. Nous avons conservé les deux objectifs afin que notre modèle bénéficie de la KD sur les entités tout en gardant son pouvoir prédictif sur le vocabulaire courant afin de pallier au phénomène d'oublis catastrophiques.

Afin de *collecter les probabilités des entités à partir d'un PLM*, nous calculons d'abord les probabilités à partir d'un PLM sur des phrases disposant d'une mention d'une entité (e_i) et d'un contexte c_i, où la mention est masquée. Nous évaluons la probabilité postérieure que chaque entité complète le masque par rapport au contexte c_i :

$$\mathcal{P}(e_i|c_i)|\forall e_i \tag{6.1}$$

Ensuite, nous estimons la probabilité de toute entité $e_l \in \mathcal{E}$ d'être pertinente pour le contexte donné c_i comme suit :

$$\hat{\mathcal{P}}(e_l|c_i, \theta_{mlm}) = \frac{\exp(mlm(e_l, c_i))}{\sum_{e_j \in \mathcal{E}} \exp(mlm(e_j, c_i))} \tag{6.2}$$

où la fonction mlm est notre prédicteur pour la tâche MLM et θ_{mlm} ses paramètres. Notez qu'idéalement, le prédicteur donnera une probabilité maximale à l'entité masquée, par exemple e_i.

3.1.2 *Extension du vocabulaire du PLM*

Nous pouvons souligner que le vocabulaire d'un PLM est composé d'un nombre limité de séquences de caractères se répétant fréquemment. Par conséquent, une entité $e_i \in \mathcal{E}$ peut être composée de m_i sous-unités de mots au sein du vocabulaire du PLM. Pour surmonter ce problème, nous avons sélectionné comme label pour l'entité e_i, une mention qui fait déjà partie du vocabulaire du PLM et alternativement la mention la plus fréquente. Ainsi, l'entité *"Rio de Janeiro"* devient *"Rio"* si cette dernière est sa mention la plus fréquente. Lorsqu'aucune mention n'a été trouvée dans le vocabulaire, nous ajoutons l'entité e_i au vocabulaire en initialisant la représentation de l'entité e_i comme la moyenne des représentations des unités de mots qui composent l'entité. Nous mettons à jour la dernière couche et ajoutons les nouveaux mots cibles :

$$embedding(e_i) = \frac{\sum_{m_j \in tokenizer(e_i)} embedding(m_j)}{|tokenizer(e_i)|} \tag{6.3}$$

3.1.3 *Distillation des entités par le modèle de représentation des KBs*

De façon symétrique, afin de collecter les probabilités des entités à partir d'un modèle de représentation KB, nous calculons les probabilités de chaque entité par rapport à la relation et à l'objet ou au prédicat du triplet en utilisant :

$$\mathcal{P}(e_i|e_j, r_k, \theta_{lp}) \tag{6.4}$$

où l'entité e_j du contexte fait partie d'un triplet existant $[(e_i, r_k, e_j)]$ ou $(e_j, r_k, e_i) \in T^+ \; \forall \; r_k \in |\mathcal{R}|, e_i, e_j \in |\mathcal{E}|$.

Nous approximons cette distribution de probabilité via l'estimateur :

$$\hat{\mathcal{P}}(e_i|e_j, r_k, \theta_{lp}) = \frac{\exp(f(e_i, r_k, e_j))}{\sum_{e_{i\prime} \in \mathcal{E}} \exp(f(e_{i\prime}, r_k, e_j))} \tag{6.5}$$

où $f(.,.,.)$ est un modèle dédié à la tâche de prédiction des liens tel que TransE (Bordes *et al.*, 2013) et θ_{lp} ses paramètres, et l'entité e_j et la relation r_k sont obtenues de la KB si le triplet existe, par exemple $(e_i, r_k, e_j) \in T^+$. Notez que la position de l'objet ou du prédicat de e_i dans le triplet n'affecte que l'ordre des premier et troisième paramètres dans $f(.,.,.)$.

3.2 *Apprentissage coopératif*

Nous détaillons ici la formulation de l'objectif coopératif de notre modèle de langage augmenté et la procédure de régularisation associée.

3.2.1 *Fonction objectif coopérative*

Notre entraînement coopératif implique la mise à jour successive des paramètres du PLM et du modèle de KB qui jouent alternativement les rôles de l'enseignant et de l'élève, comme suggéré dans des travaux antérieurs : Zhang *et al.* (2018); Sourty *et al.* (2020); Guo *et al.* (2020). Nous formulons l'objectif d'apprentissage mutuel entre les tâches de prédiction de liens et de MLM comme suit :

$$\mathcal{L}^{kd} = \mathcal{D}(\hat{\mathcal{P}}(e_j|e_i, r_k, \theta_{lp}), \hat{\mathcal{P}}(e_j|c_i, \theta_{mlm})) \tag{6.6}$$

où θ_{lp} et θ_{mlm} sont les paramètres d'un modèle de représentation de KB et d'un PLM, respectivement. e_i est une entité $\in \mathcal{E}$ qui est mentionnée et cachée dans le contexte c_i. Nous utilisons la fonction de divergence de Kullback-Leibler comme mesure de distance $\mathcal{D}$.

3.2.2 *Normalisation de l'objectif coopératif*

Afin de combiner équitablement les objectifs de la KB et du PLM et de stabiliser l'apprentissage mutuel, nous avons appliqué la normalisation de la fonction objectif proposée dans Zoph *et al.* (2020) pour formuler l'objectif global (c'est-à-dire la modélisation du langage masqué plus la distillation des connaissances) de chacun de nos modèles améliorés. La fonction de perte de notre PLM augmenté est :

$$\mathcal{L}_{plm} = \frac{1}{1+\alpha_{mlm}} \left(\mathcal{L}^{kd} + \alpha_{mlm} \frac{\overline{\mathcal{L}^{kd}}}{\overline{\mathcal{L}_{mlm}}} \mathcal{L}_{mlm} \right) \tag{6.7}$$

où $\overline{\mathcal{L}^{kd}}$ et $\overline{\mathcal{L}_{mlm}}$ désignent les moyennes exponentielles de l'objectif de distillation des connaissances et de modélisation du langage masqué enregistrées précédemment.

De même, la fonction objectif de notre modèle de représentation de KB augmenté est :

$$\mathcal{L}_{kb} = \frac{1}{1+\alpha_{lp}} \left(\mathcal{L}^{kd} + \alpha_{lp} \frac{\overline{\mathcal{L}^{kd}}}{\overline{\mathcal{L}_{lp}}} \mathcal{L}_{lp} \right) \tag{6.8}$$

où $\overline{\mathcal{L}_{lp}}$ désigne les moyennes exponentielles de l'objectif de prédiction des liens enregistrées précédemment.

4 Cadre expérimental

4.1 *Configuration coopérative et baselines*

Nous avons mené une série d'expériences pour évaluer les performances intrinsèques et extrinsèques de notre modèle par rapport aux baselines. Nous présentons ci-dessous le jeu de données utilisé, les configurations de nos expériences et nos principaux résultats.

4.1.1 *Stratégies de distillation*

La configuration de notre PLM augmenté est désignée par `Cooptiv`. Dans cette configuration, les modèles de PLM et de représentation de KB sont mis à jour par distillation des connaissances sur leurs tâches respectives, c'est-à-dire MLM via l'objectif 6.7 et sur la prédiction des liens via l'objectif 6.8. Nous comparons notre modèle à deux configurations de base distinctes :

— `Vanilla` : Les deux modèles sont entraînés sur leurs tâches respectives, c'est-à-dire le MLM et la prédiction des liens. La distillation des connaissances n'est pas impliquée dans cette stratégie.
— `Knowldg` : Les deux modèles sont entraînés sur leurs tâches respectives, c'est-à-dire le MLM et la prédiction de liens. Seul le MLM bénéficie de la distillation via l'objectif 6.7.

Ces configurations partagent les mêmes hyperparamètres et ont été entraînées à l'aide de deux PLM distincts :

— `PLM-A` : Distil BERT (Sanh *et al.*, 2019), un modèle à base d'un *transformer* de 44 millions de paramètres ;
— `PLM-B` : BERT-base (Devlin *et al.*, 2019), un autre modèle à base d'un *transformer* de 110 millions de paramètres.

Nous avons entraîné tous les modèles avec l'optimiseur Adam, un taux d'apprentissage de 5e-8 et une taille de batch de 32. En ce qui concerne la représentation des entités, nous avons utilisé le modèle TransE et nous avons fixé la dimension des représentations à 500 pour les relations et les entités. Nous avons fixé le taux d'apprentissage à $5e-6$ et utilisé l'optimiseur Adam et fixé la taille des mini-batchs à 512. Pour chaque triplet positif, nous avons généré 512 triplets corrompus en suivant l'objectif adversarial de prédiction des liens définie par Sun *et al.* (2019b) avec un paramètre de marge γ fixé à 6. De plus, nous suivons Zoph *et al.* (2020) pour définir le taux de décroissance de la moyenne mobile exponentielle à 0,9997

dans le cadre des équations 6.7 et 6.8. Enfin, nous avons fixé les paramètres dédiés à la normalisation de l'objectif, α_{lp} et α_{mlm}, à 0,5.

	Dataset	
	FB15K-237	**Wikipedia**
# Entités	14,541	12,516
# Relations	237	-
# Entraînement	272,115	8,000,000
# Validation	17,535	30,000
# Test	20,466	30,000
Couverture des entités	-	86.07
Couverture des triplets	-	74.90

TABLEAU 6.1 – Statistiques de nos corpus de pré-entraînement dédiés aux tâches de MLM et de prédiction des liens.

4.1.2 *Pré-traitement de l'ensemble de données*

Nous avons utilisé le graphe de connaissances FB15K-237 comme notre corpus principal. Les statistiques du KB sont présentées dans le tableau 6.1 (colonne de gauche). Cependant, comme un corpus de texte est nécessaire, nous avons aligné les entités FB15K-237 et leurs mentions dans Wikipédia en utilisant des liens hypertexte dans le but d'effectuer conjointement les tâches de MLM et de prédiction des liens. Nous avons sélectionné 8 millions de phrases de Wikipédia qui mentionnent au moins une entité de FB15K-237. Les statistiques des corpus de textes sont présentées dans le tableau 6.1 (colonne de droite). Notre échantillon de Wikipédia présente un taux de couverture significatif des entités FB15K-237 avec au moins une mention de 86,07% des entités et 74,90% des triplets Freebase (ensembles de formation, de validation et de test combinés). Nous avons échantillonné 60 000 phrases pour construire un ensemble de validation et un ensemble de test afin d'effectuer des évaluations intrinsèques telles que l'évaluation de la perplexité de nos PLM augmentés. Pour assurer la couverture des entités utilisées, nous avons ajouté 12 230 entités au vocabulaire des PLM utilisés.

4.2 *Évaluation intrinsèque*

Dans le cadre de l'évaluation intrinsèque de nos modèles augmentés, nous mesurons la capacité des modèles de langage à traiter le vocabulaire, les entités communes et rares dans le cadre de la tâche de modélisation du langage masqué.

4.2.1 *Mesure de perplexité*

Tous les modèles et configurations ont été évalués en mesurant la mesure de perplexité des modèles de langage masqué (PPL) sur notre ensemble de test Wikipédia. Notez que comme l'information sur le mot masqué est connue, nous pouvons calculer la perplexité en considérant si le mot est une entité ou non. De plus, nous avons évalué les performances sur l'ensemble de données FB15k-237 en utilisant des mesures de prédiction de liens standard. Les résultats sur les trois configurations utilisant les deux PLMs sont présentés dans le Tableau 6.2.

	Modélisation du langage masqué		**Prédiction des liens**			
Model	**PPL Entité**	**PPL**	**HITS@1**	**HITS@3**	**HITS@10**	**MRR**
Vanilla PLM-A	10.12	7.55	**22.53**	**36.27**	**52.15**	**0.32**
Knowldg PLM-A	**8.36**	**7.37**	**22.53**	**36.27**	**52.15**	**0.32**
Cooptiv PLM-A	8.38	7.41	21.02	34.58	50.21	0.30
Vanilla PLM-B	7.81	**6.02**	**22.53**	**36.27**	**52.15**	**0.32**
Knowldg PLM-B	**7.28**	6.40	**22.53**	**36.27**	**52.15**	**0.32**
Cooptiv PLM-B	7.31	6.34	20.95	34.55	50.19	0.30

TABLEAU 6.2 – Évaluation intrinsèque de notre PLM augmenté et de notre modèle de représentation de KB.

Comme on peut le constater, `Cooptiv PLM-A` a obtenu un score de 7,41 en PPL contre 7,55 pour `Vanilla PLM-A`. En ce qui concerne `PLM-B`, les deux stratégies `Knowldg` et `Cooptiv` dégradent légèrement la mesure de perplexité 6,40 & 6,34 contre 6,02 pour `Vanilla`. Dans l'ensemble, les stratégies `Cooptiv` et `Knowldg` conservent la capacité des PLM à traiter les mots du vocabulaire courants. Nous avons calculé la métrique "PPL Entité" en mesurant la perplexité exclusivement sur les mentions des entités de notre KB dans Wikipedia. Les deux stratégies `Cooptiv` et `Knowldg` diminuent la perplexité des entités par rapport à `Vanilla` et attribuent une plus grande probabilité à la vérité terrain lorsqu'il s'agit d'une entité faisant partie de FB15K-237. `Cooptiv PLM-A` améliore la perplexité sur les entités par rapport à `Vanilla PLM-A` (8,38 contre 10,12), et `Cooptiv PLM-B` obtient un score

de 7,31 contre 7,81 pour `Vanilla PLM-B`. Nous vérifions la cohérence du modèle TransE via l'évaluation de la prédiction des liens. Pour ce faire, nous avons également mesuré les scores de prédiction de liens de nos modèles de représentation de KB en fonction de la stratégie avec les triplets de test FB15K-237. Il convient de noter que les modèles TransE ne bénéficient pas de la distillation des connaissances, mais qu'ils ne donnent pas non plus de résultats aberrants : Les modèles TransE entraînés en binôme avec `Cooptiv PLM-A` mènent à une diminution de $-4,66\%$ sur la métrique HITS@3 et `Cooptiv PLM-B` mène à une réduction de $-4,74\%$. TransE n'est pas excessivement biaisé en faveur du modèle de langue malgré le fait que nous ayons fixé le facteur de normalisation α_{lp} à $0,5$ (voir l'équation 6.8) et qu'il accorde de l'importance aux softs labels de `PLM-A` et `PLM-B`. Nous pensons que les natures différentes et les objectifs distincts entre les PLMs et les KB embeddings conduisent à l'absence d'amélioration sur la tâche de prédiction de liens. En outre, Stanton *et al.* (2021) présente le compromis entre la complexité de l'optimisation et la qualité des données de distillation, c'est-à-dire qu'un élève qui reproduit les prédictions d'un professeur via la distillation de connaissances ne conduit pas systématiquement à une amélioration.

4.2.2 *Modélisation des entités rares*

Nous avons également mesuré la capacité de notre PLM augmenté à retrouver une entité masquée en fonction de sa fréquence d'apparition dans le corpus d'entraînement (c'est-à-dire le seuil). Sur la base des résultats présentés dans le tableau 6.3, nous montrons que moins une entité est mentionnée, plus il sera difficile pour le modèle linguistique de la retrouver. Par exemple, 337 paragraphes de notre corpus de test mentionnent une entité référencée moins de 150 fois (colonne seuil) dans notre corpus d'entraînement. Cette évaluation reflète la difficulté des modèles de langage à s'adapter à des entités peu fréquentes ou à de nouveaux domaines. Les modèles `Vanilla PLM-A` et `Vanilla PLM-B` ne classent que 17,46% des entités masquées (avec un seuil < 150) dans les 100 premières entités. La distillation de la base de connaissances améliore considérablement la capacité des modèles à trouver des entités rarement mentionnées dans Wikipédia. Les stratégies `Cooptiv` et `Knowldg` surpassent systématiquement la stratégie `Vanilla` pour les deux options `PLM-A` et `PLM-B` sur la métrique Precision@100. Le modèle de langage améliore la représentation interne de ses entités via des softs labels sans nécessiter de nombreux exemples explicites dans le corpus d'apprentissage. Par exemple, nous observons une amélioration relative de $33,28\%$ ($12,48\%$ respectivement) sur la métrique Precision@10 entre le modèle `Vanilla PLM-A` (`Vanilla PLM-B` respectivement) et `Cooptiv PLM-A` (`Cooptiv PLM-B` respectivement) pour les entités mentionnées moins de 50 fois dans notre sous-ensemble de Wikipedia.

		Modélisation du langage masqué		
Seuil	**Model**	**P@1**	**P@10**	**P@100**
50	*Vanilla PLM-A*	**2.06**	6.19	16.49
	Knowldg PLM-A	1.03	**8.25**	**19.59**
	Cooptiv PLM-A	1.03	**8.25**	**19.59**
	Vanilla PLM-B	2.06	8.25	18.56
	Knowldg PLM-B	**4.12**	**9.28**	**21.65**
	Cooptiv PLM-B	3.09	**9.28**	**21.65**
150	*Vanilla PLM-A*	**2.12**	7.67	17.46
	Knowldg PLM-A	1.32	7.67	19.84
	Cooptiv PLM-A	1.32	7.67	**20.11**
	Vanilla PLM-B	**2.65**	6.88	17.46
	Knowldg PLM-B	1.85	7.94	20.63
	Cooptiv PLM-B	1.59	**8.20**	**20.90**
300	*Vanilla PLM-A*	**2.82**	**10.06**	24.54
	Knowldg PLM-A	2.45	10.43	26.38
	Cooptiv PLM-A	2.70	9.94	**26.87**
	Vanilla PLM-B	**2.82**	8.10	22.33
	Knowldg PLM-B	2.33	9.45	25.89
	Cooptiv PLM-B	2.21	**9.69**	**26.01**

TABLEAU 6.3 – Précision de la modélisation du langage masqué avec un focus sur les entités rares. Le seuil représente la limite supérieure de la fréquence de l'entité cible dans notre corpus Wikipedia.

4.3 *Évaluation extrinsèque du modèle de langue sur la tâche de slot filling*

Nous présentons ici nos expériences pour saisir l'impact de notre stratégie coopérative sur la tâche de slot filling.

4.3.1 *Ensembles de données et baselines*

Nous avons évalué tous les modèles sur deux datasets dédiés à la tâche de slot filling, T-REx (Elsahar *et al.*, 2018) et zsRE (Levy *et al.*, 2017). Comme proposé dans Petroni *et al.* (2021), nous avons collecté 2284168 paires de questions et de réponses pour T-REx et 197620 paires pour zsRE. Comme baselines, nous avons opté pour BERT + DPR, BART + DPR, et RAG fournis par le classement KILT. BERT + DPR est un pipeline composé d'un retriever responsable de filtrer les documents pertinents et d'un modèle d'extraction de réponses aux questions. La baseline BART + DPR est performante et bénéficie de son grand nombre de paramètres par rapport à BERT et de la capacité du lecteur à générer des réponses héritées des modèles seq2seq. RAG est un pipeline de bout en bout entraîné sur la tâche de slot filling et basé sur un retriever DPR et un reader BART. Enfin, DensePhrases^{10k} s'appuie sur le modèle SpanBERT-base-cased (Joshi *et al.*, 2020).

4.3.2 *Configuration du modèle de slot filling*

Nous avons entraîné nos modèles à l'aide d'un objectif d'extraction de réponses aux questions (QA) afin de compléter les slots. Pour construire l'ensemble d'entraînement des modèles de QA, nous avons aligné les requêtes telles que (e_i, s_k) et les passages de Wikipédia qui ont au moins une des réponses attendues pour compléter les slots $\in \{e_j^0, e_j^1, .., e_j^n\}$. Nous avons entraîné les modèles de QA sur le dataset T-REx pour une seule epoch avec l'optimiseur AdamW, avec un taux d'apprentissage fixé à $2e-5$ et une taille de mini-batch de 16. Pour zsRE, nous nous sommes appuyés sur cinq epochs et l'optimiseur AdamW avec un taux d'apprentissage de $2e-5$. Nous avons ajouté une régularisation à nos modèles pour les deux modèles de slot filling en fixant le coefficient de décroissance des poids d'AdamW à $0,01$.

Lors de l'inférence, nous avons commencé par diviser en paragraphes la source de connaissances composée de $5,9$ millions de documents partagés par KILT. Cela représente plus de 110 millions de paragraphes que nous avons indexés avec BM25. Ensuite, nous avons filtré les 200 paragraphes les plus pertinents en suivant le cadre de retriever-reader pour chaque requête. Nous avons récupéré les documents en utilisant le titre de la page Wikipedia et le contenu du paragraphe pour T-REx. Pour zsRE, nous avons utilisé uniquement le titre de la page Wikipedia, qui contient souvent les entités sujettes. Nous avons sélectionné les champs utilisés par le retriever en évaluant l'ensemble du pipeline retriever-reader sur le jeu de données de validation de T-REx et zsRE. Nous avons finalement extrait la réponse la plus probable parmi les 200 premiers paragraphes récupérés pour chaque requête avec nos PLMs.

4.3.3 *Métriques*

Nous avons évalué nos modèles à l'aide du benchmark KILT. KILT évalue les performances d'un modèle sur 1) sa capacité à retrouver des preuves (R-PREC, Recall@5), 2) la précision des candidats proposés par le système (Précision, F1), et 3) une combinaison des deux métriques de recherche et de précision (KILT-AC, KILT-F1). KILT-AC et KILT-F1 correspondent à une Précision et F1 pour lesquelles une réponse est correcte si le document qui a permis de la trouver est classé premier par le système. Par conséquent, KILT-AC <= Précision et KILT-F1 <= F1. Les métriques KILT-AC et KILT-F1 mettent l'accent sur l'interprétabilité.

	KILT-AC		KILT-F1		R-Prec		Précision		F1		Recall@5	
Model	T-REx	zsRE	T-REx	zsRE	T-REx	zsRE	T-REx	zsRE	T-REx	zsRE	T-REx	zsRE
`Vanilla PLM-A`	34.69	31.93	37.57	35.06	46.50	61.65	46.72	33.66	52.69	37.47	51.07	63.37
`Knowldg PLM-A`	34.96	32.23	37.77	35.15	46.90	59.68	46.36	34.65	51.86	38.18	50.89	62.04
`Cooptiv PLM-A`	**36.68**	34.13	**39.56**	37.22	**48.08**	61.33	49.04	36.22	54.61	40.33	**51.86**	63.85
`Vanilla PLM-B`	33.08	31.05	35.96	35.48	44.58	59.31	45.5	36.09	51.02	40.61	49.24	63.32
`Knowldg PLM-B`	32.18	28.79	35.01	32.54	43.94	57.20	44.44	32.64	50.77	37.43	49.20	60.53
`Cooptiv PLM-B`	34.38	35.32	37.34	39.55	46.56	**63.18**	46.42	38.54	51.88	44.03	50.38	**66.51**
BERT + DPR (Petroni et al., 2021)	-	4.47	-	27.09	-	40.11	-	6.93	-	37.28	-	40.11
BART + DPR (Petroni et al., 2021)	11.12	18.91	11.41	20.32	13.26	28.90	59.16	30.43	62.76	34.47	17.04	39.21
RAG (Lewis et al., 2020b; Petroni et al., 2021)	23.12	36.83	23.94	39.91	28.68	53.73	**59.20**	44.74	**62.96**	49.95	33.04	59.52
DensePhrases 10^k *(Lee et al., 2021)*	27.84	**41.34**	32.34	**46.79**	37.62	57.43	53.90	**47.42**	61.74	**54.75**	40.07	60.47

Tableau 6.4 – Performances extrinsèque sur les tâches de slot filling avec le jeu de données KILT. Nous présentons les résultats des trois stratégies distinctes, à savoir `Vanilla`, `Knowldg`, et `Cooptiv` pour `PLM-A` et `PLM-B`.

4.3.4 *Résultats et discussion*

Le tableau 6.4 résume les résultats de nos modèles sur la tâche de slot filling : les stratégies `Vanilla`, `Knowldg`, et `Cooptiv` pour les deux modèles `PLM-A` et `PLM-B`. Dans l'ensemble, notre modèle `Cooptiv PLM-A` atteint des performances compétitives sur le jeu de données T-REx par rapport à DensePhrases avec une amélioration relative de 31,75 % sur la métrique KILT-AC, 22,32 % sur KILT-F1, 27,80 % sur R-Prec et 29,42 % sur Recall@5. Nos modèles se fient davantage aux documents contenant la vérité terrain : Les stratégies `Vanilla, Knowledg, Cooptiv` pour les deux modèles `A` et `B` améliorent systématiquement R-Prec et Recall@5 par rapport à Petroni *et al.* (2021); Lee *et al.* (2021); Lewis *et al.* (2020b). La stratégie `Cooptiv` surpasse systématiquement ses homologues `Vanilla` et `Knowldg` sur toutes les métriques (KILT-AC, KILT-F1, Précision, F1, et Recall@5), pour les deux modèles `PLM-A` et `PLM-B`, démontrant la valeur de notre augmentation via le pré-entraînement coopératif.

Sujet	Type	Requête	Golds	Model	Réponses
Album	Q^+	William Shakespeare [SEP] genre	early modern english theatre, .., english renaissance drama	*Vanilla PLM-A*	shakespeare, sonneteers, comedies, dramatists, dramatist
				Cooptiv PLM-A	**english renaissance**, tragedy, comedies, parodying, dramatist
		Phil Nimmons [SEP] occupation	composer	*Vanilla PLM-A*	architect, technologist, jazz musician, bandleaders, bullet
				Cooptiv PLM-A	architect, technologist, **composer**, bandleaders, jazz musician
	Q^-	Sweet Memories [SEP] genre	romance genre, romance film, romantic film	*Vanilla PLM-A*	**romance film**, romantic drama, country artist, country
				Cooptiv PLM-A	country artist, adult contemporary, country tracks, willie nelson, country
		music manuscript [SEP] instance of	manuscripts, ms, manuscript, .., manuscript books	*Vanilla PLM-A*	video game, **manuscript**, musical terminology, software, library
				Cooptiv PLM-A	video game, library, terminology, musical terminology, software
Australia	Q^+	New York State Route 119 [SEP] country	united states of america, u. s. a., u.s., ..,the us	*Vanilla PLM-A*	utah, new york, nevada, washington, u.s. state of washington. state of utah
				Cooptiv PLM-A	**united states**, utah, washington, new york, u.s. state of washington
		New York State Route 316 [SEP] country	united states of america, u. s. a., u.s., ..,the us	*Vanilla PLM-A*	georgia, ohio, new york, u.s. state of georgia, pickaway county
				Cooptiv PLM-A	**united states**, georgia, ohio, new york, south bloomfield
	Q^-	Allied invasion of Sicily [SEP] country	it, ita, italian republic, italy, italia	*Vanilla PLM-A*	malta, **italy**, greece, canada, kingdom of italy, tunisia
				Cooptiv PLM-A	canada, malta, greece, sicily, kingdom of italy
		subregion of Finland [SEP] country	land of thousand lakes, finnland, finlande, .., finlandia	*Vanilla PLM-A*	portugal, uganda, poland, colombia, **finland**
				Cooptiv PLM-A	uganda, portugal, poland, united nations, colombia

TABLEAU 6.5 – Les premières réponses sur le jeu de données T-REx récupérées par `Vanilla PLM-A` et `Cooptiv PLM-A` ordonnées par vraisemblance. Q^+ indique les requêtes pour lesquelles notre PLM amélioré est meilleur que son homologue vanille et vice versa pour Q^-.

Nous reportons dans la figure 6.2 l'amélioration relative de la métrique de précision fournie par le modèle `Cooptiv PLM-A` par rapport à son homologue Vanilla. Nous avons construit les 41 sujets en suivant la procédure définie par la librairie Python BERTopic (Grootendorst, 2020) en utilisant le Sentence Transformer pré-entraîné "all-MiniLM-L6-v2" (Reimers et Gurevych, 2019). BERTopic s'appuie sur un TF-IDF basé sur la classe pour extraire le mot le plus représentatif comme descripteur de sujet pour chaque cluster. Nous pouvons constater que le modèle `Cooptiv PLM-A` obtient de meilleurs résultats sur les sujets *géographie, science*, et *éducation* par rapport au modèle `Vanilla PLM-A` avec une augmentation relative de la précision de 25%. Le modèle *Cooptiv PLM-A* améliore considérablement les résultats sur les clusters *1. death, 2. Australia*, et *3. school*, et réduit les performances sur les clusters *4. origin, 5. album*, et *6. platform*. Les sujets pour lesquels nous observons une amélioration font référence à des entités surreprésentées dans les triplets d'apprentissage de notre base de connaissances. Les sujets améliorés ont tendance à mentionner plus d'entités de notre base de connaissances. 2931 triplets d'apprentissage de FB15K-237 référençant directement le sujet *"Australia"* contre 66 référençant le sujet *"plateform"*. Les entités appartenant aux thèmes 1, 2 et 3 sont surreprésentées dans notre corpus Wikipedia. Par exemple, 1,37% des articles Wikipédia que nous avons utilisés pour effectuer notre procédure d'amélioration coopérative mentionne une entité du thème *"death"* contre 0,45% pour le thème *"platforme"*. 17,58% des mots du thème *"schools"* récupérés par le TF-IDF font partie des entités de FB15K-237 contre 2,62% pour le thème *"origin"*. Le tableau 6.5 donne un aperçu des prédictions de notre PLM augmenté `Cooptiv PLM-A`, qui obtient les meilleurs résultats sur l'ensemble des données de validation de T-REx par rapport à son homologue `Vanilla PLM-A` (le test n'est disponible que via le classement officiel). Dans ce tableau, pour les sujets *"Album"* et *"Australia"* (colonne Topic, voir figure 6.2), nous indiquons les cinq meilleures réponses de chaque modèle (colonnes Réponses) pour plusieurs requêtes (colonne Requête) de l'ensemble de données de validation T-REx. Nous distinguons les exemples pour lesquels notre modèle augmenté fournit la réponse attendue (colonne Golds) avec le type Q^+ (colonne Type) des exemples pour lesquels la version standard est correcte avec le type Q^-. Le premier exemple, *[]William Shakespeare [SEP] genre]* fait partie de l'ensemble de données de validation T-REx. Notre modèle amélioré, `Cooptiv`, récupère la vérité terrain *théâtre de la renaissance anglaise* et propose avec succès l'entité *tragédie* de FB15K-237. Les entités géographiques sont surreprésentées dans les triplets de FB15k-237; plus de 20% des triplets de FB15K-237 font référence à une zone géographique. Ces entités géographiques font référence à un *lieu de naissance* ou de *mort* (thème death), à la localisation d'une *université* (thème schools), ou, plus globalement, à des infrastructures nationales (thème Australia). Grâce à la distillation de la KB, `Cooptiv PLM-A` développe une meilleure représentation interne des entités géographiques et répond *united states* à la requête [*New*

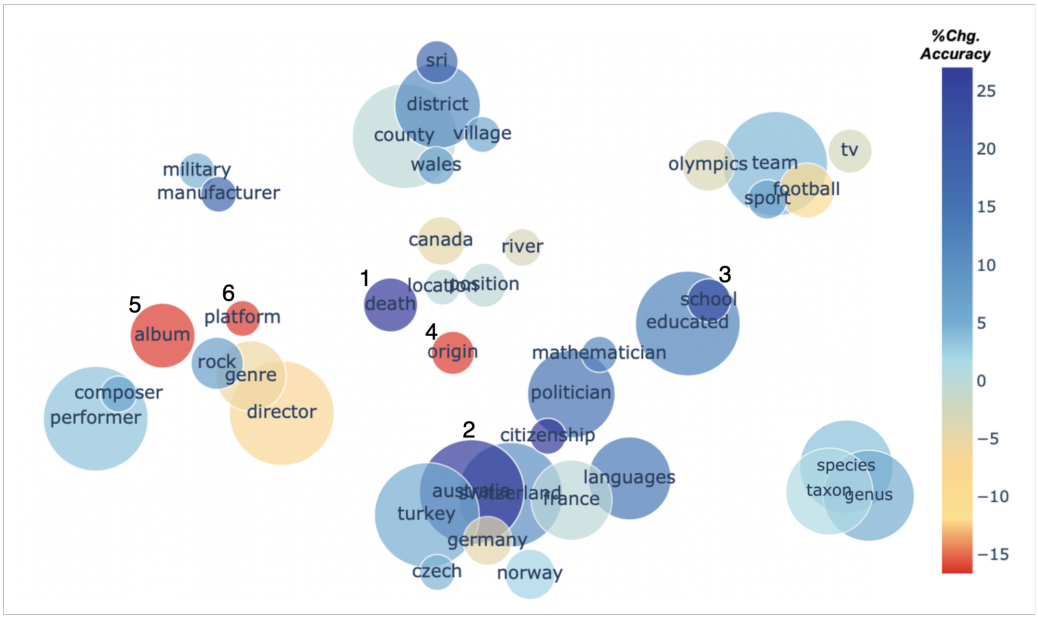


FIGURE 6.2 – Améliorations observées de la précision avec le modèle `Cooptiv PLM-A` sur le jeu de données T-REx par thème par rapport au modèle `Vanilla PLM-A`. La taille des clusters est proportionnelle au nombre d'échantillons qu'ils incluent. Les 3 meilleures et les 3 pires améliorations/thèmes sont numérotés de 1 à 6.

York State Route 199 [SEP] country] alors que son homologue `Vanilla` liste les villes et régions.

5 Bilan

Dans cette étude, nous avons présenté une stratégie d'enrichissement des PLM basée sur la distillation coopérative des connaissances. Notre proposition repose sur l'alignement de deux tâches existantes, la prédiction de liens et la MLM, par l'utilisation de pseudo-étiquettes et l'apprentissage mutuel. Nous tirons parti des corpus textuels mentionnant des entités pour estimer et transférer les probabilités d'entités d'une représentation KB à un MLM et vice versa. Nous avons mené des évaluations expérimentales intrinsèques et extrinsèques de nos modèles sur des jeux de données standards. L'évaluation intrinsèque montre que les deux espaces appris sont complémentaires et que les softs labels des KBs permettent aux PLMs d'améliorer leur représentation interne des entités. Enfin, nous avons montré empiriquement que sur deux jeux de données standards pour des tâches orientées connaissances (T-REx et zsRE) nos PLM enrichis sont plus efficaces que leurs homologues classiques (entre 1,38 et 4,07 en termes de KILT-F1) et ce, selon la plupart des métriques. Un examen plus approfondi des résultats a montré que la plupart des thèmes de l'ensemble de données bénéficient de la représentation de notre modèle avec une amélioration plus faible sur les thèmes qui se rapportent davantage aux entités.

Partie III

CONCLUSION

CONCLUSION

Synthèse des contributions

L'un des principaux défis de la représentation des connaissances est l'intégration de connaissances structurées provenant de sources multiples. Notre travail s'est concentré sur ce défi en étudiant les différents modèles et méthodes de construction de représentations de graphes de connaissances, et en étudiant le potentiel des techniques de distillation des connaissances. Nos recherches ont montré que les procédures d'apprentissage mutuel constituent une approche prometteuse pour fusionner des espaces de connaissances distinctes et pour créer des représentations de bases de connaissances multiples. Nous avons proposé un nouveau cadre appelé *KD-MKB*pour l'apprentissage des représentations des entités et des relations de multiples bases de connaissances. Notre approche est basée sur le cadre de distillation de connaissances et d'apprentissage mutuel, qui permet aux modèles de bases de connaissances d'"apprendre" les uns des autres et d'améliorer leurs performances dans le cadre de la tâche de prédiction de liens. Nous formalisons l'inférence des entités et des relations entre les bases de connaissances comme un objectif de distillation sur les distributions de probabilité postérieures des connaissances alignées. Nos expériences démontrent l'efficacité de notre approche sur la tâche de prédiction de liens par rapport aux stratégies de distillation existantes. Nous avons mis en évidence l'importance du mutual learning entre les modèles de représentations des KBs pour transmettre efficacement la connaissance. Nous pensons que le cadre que nous proposons peut être développé et appliqué à d'autres domaines qui s'appuient sur de grandes quantités de données structurées.

La seconde contribution de cette thèse est dédiée à l'intégration de deux espaces distincts pour la tâche de slot filling : le corpus textuel et la base de connaissances. Nous proposons un cadre coopératif de distillation des connaissances qui aligne la tâche de MLM des modèles de langage avec l'objectif de prédiction de liens des modèles de représentation de KB. Le corpus textuel fournit des informations non structurées sous de langage naturel, tandis que la base de connaissances contient des informations structurées sur les entités et les relations sous la forme de triplets. La méthode que nous proposons utilise les forces des deux espaces en incorporant des informations riches sur les entités provenant de la base de connaissances lors

du processus de pré-entraînement des PLMs. Nous sommes en mesure d'améliorer les performances de nos modèles sur la tâche extrinsèque de slot filling en entraînant les PLMs à produire des softs labels de qualité pour les entités qui sont susceptibles d'apparaître dans le texte.

Perspectives

Généralisation du modèle KD-MKB

Il est important de noter que dans notre travail, nous avons évalué *KD-MKB* avec TransE (Bordes *et al.*, 2013). Il serait pertinent d'explorer d'autres modèles de représentation des connaissances (Sun *et al.*, 2019b; Trouillon *et al.*, 2016) pour évaluer les performances de notre procédure. En outre, nous pourrions étudier l'impact de différentes stratégies d'alignement sur les performances de notre approche, en particulier pour les bases de connaissances de différentes tailles et de différentes sources. Ces perspectives pourraient contribuer à une meilleure compréhension de l'efficacité et de l'applicabilité de notre approche *KD-MKB* pour la représentation des connaissances à grande échelle.

Distillation et bases de connaissances multimodales

Avec la disponibilité croissante de diverses modalités de données, y compris le texte, les images, les vidéos, il est de plus en plus nécessaire de développer des modèles capables de représenter efficacement ces sources d'information complémentaires dans les KBs (Pezeshkpour *et al.*, 2018). L'une des pistes de travail potentielles est d'explorer la manière dont les techniques de distillation des connaissances peuvent être adaptées aux bases de connaissances multimodales. Cela pourrait impliquer le développement de méthodes d'alignement et d'intégration de différents modèles, ainsi que l'exploration de nouvelles approches pour la construction de graphes de connaissances qui capturent les relations entre les données de différentes natures.

Apprentissage mutuel des PLMs et des KBs

Dans nos expériences, nous avons évalué la distillation des bases de connaissances vers les PLMs avec des échantillons de bases de connaissances en particu-

lier *FB15K237* (Toutanova *et al.*, 2015). Bien que nous ayons montré l'efficacité de notre approche via l'évaluation extrinsèque sur la tâche de slot-filling, il serait pertinent d'évaluer cette approche avec une base de connaissances réelle qui dispose d'une couverture plus large des entités. Cela pourrait nous permettre de mieux comprendre les avantages et les limitations de notre approche pour la représentation des connaissances à grande échelle et pour une variété d'applications du monde réel. Par exemple, une base de connaissances telle que Wikidata5M (Wang *et al.*, 2021b) pourrait être utilisée pour renforcer notre modèle augmenté par les bases de connaissances.

Applications

Malgré les progrès récents dans le domaine du traitement du langage naturel et de la représentation des connaissances, il existe encore un fossé important dans la disponibilité de logiciels libre de droits qui intègrent profondément les applications bases de connaissances et les modèles de langage de l'état de l'art. Bien qu'il existe plusieurs outils spécialisés dans l'un ou l'autre domaine (Wolf *et al.*, 2020; Sourty *et al.*, 2020, 2022a,b), il est nécessaire d'adopter une approche plus globale qui combine les points forts des deux domaines et qui s'appuie sur les publications scientifiques de l'état de l'art.

BIBLIOGRAPHIE

Jimmy Ba et Rich Caruana : Do deep nets really need to be deep ? *Advances in neural information processing systems*, 27, 2014.

Jimmy Lei Ba, Jamie Ryan Kiros et Geoffrey E Hinton : Layer normalization. *arXiv preprint arXiv :1607.06450*, 2016.

Stefano Baccianella, Andrea Esuli et Fabrizio Sebastiani : SentiWordNet 3.0 : An enhanced lexical resource for sentiment analysis and opinion mining. *In Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, mai 2010. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2010/pdf/769_Paper.pdf.

Ivana Balazevic, Carl Allen et Timothy Hospedales : TuckER : Tensor factorization for knowledge graph completion. *In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5185–5194, Hong Kong, China, novembre 2019. Association for Computational Linguistics. URL https://aclanthology.org/D19-1522.

Krisztian Balog : *Populating Knowledge Bases*, pages 189–222. Springer International Publishing, Cham, 2018. ISBN 978-3-319-93935-3. URL https://doi.org/10.1007/978-3-319-93935-3_6.

Krisztian Balog et Tom Kenter : Personal knowledge graphs : A research agenda. *In Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval*, ICTIR '19, page 217–220, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450368810. URL https://doi.org/10.1145/3341981.3344241.

Marco Baroni, Georgiana Dinu et Germán Kruszewski : Don't count, predict ! a systematic comparison of context-counting vs. context-predicting semantic vectors. *In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 238–247, 2014.

Hannah Bast, Buchhold Björn et Elmar Haussmann : Semantic search on text and knowledge bases. *Foundations and Trends in Information Retrieval*, 10(2-3):119–271, 2016.

Yoshua Bengio, Aaron Courville et Pascal Vincent : Representation learning : A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

Yoshua Bengio, Réjean Ducharme et Pascal Vincent : A neural probabilistic language model. *Advances in neural information processing systems*, 13, 2000.

Tim Berners-Lee, James Hendler et Ora Lassila : The semantic web. *Scientific American*, 284(5):34–43, 2001. ISSN 00368733, 19467087. URL http://www.jstor.org/stable/26059207.

Michele Bevilacqua et Roberto Navigli : Breaking through the 80% glass ceiling : Raising the state of the art in word sense disambiguation by incorporating knowledge graph information. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2854–2864, Online, juillet 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.acl-main.255.

Peter Bloem : Transformers from scratch, 2019. URL http://peterbloem.nl.

Piotr Bojanowski, Edouard Grave, Armand Joulin et Tomas Mikolov : Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5:135–146, 2017.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge et Jamie Taylor : Freebase : a collaboratively created graph database for structuring human knowledge. *In Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250, 2008.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston et Oksana Yakhnenko : Translating embeddings for modeling multi-relational data. *In* C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani et K. Q. Weinberger, éditeurs : *Advances in Neural Information Processing Systems 26*, pages 2787–2795. Curran Associates, Inc., 2013. URL http://papers.nips.cc/paper/5071-translating-embeddings-for-modeling-multi-relational-data.pdf.

Leo Breiman : Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell *et al.* : Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Cristian Buciluǎ, Rich Caruana et Alexandru Niculescu-Mizil : Model compression. *In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541, 2006.

Muhao Chen, Yingtao Tian, Kai-Wei Chang, Steven Skiena et Carlo Zaniolo : Co-training embeddings of knowledge graphs and entity descriptions for cross-lingual entity alignment. *In Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, page 3998–4004. AAAI Press, 2018. ISBN 9780999241127.

Muhao Chen, Yingtao Tian, Mohan Yang et Carlo Zaniolo : Multilingual knowledge graph embeddings for cross-lingual knowledge alignment. *In Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, page 1511–1517. AAAI Press, 2017. ISBN 9780999241103.

Kevin Clark, Minh-Thang Luong, Urvashi Khandelwal, Christopher D. Manning et Quoc V. Le : BAM ! born-again multi-task networks for natural language understanding. *In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5931–5937, Florence, Italy, juillet 2019. Association for Computational Linguistics. URL https://aclanthology.org/P19-1595.

Kevin Clark, Minh-Thang Luong, Quoc V Le et Christopher D Manning : Electra : Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv :2003.10555*, 2020.

Stephen Clark : Vector space models of lexical meaning. *Handbook of Contemporary Semantics–second edition. Wiley-Blackwell*, 2012.

Leyang Cui, Sijie Cheng, Yu Wu et Yue Zhang : On commonsense cues in BERT for solving commonsense tasks. *In Findings of the Association for Computational Linguistics : ACL-IJCNLP 2021*, pages 683–693, Online, août 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.findings-acl.61.

Daniel Daza, Michael Cochez et Paul Groth : Inductive entity representations from text via link prediction. *In Proceedings of the Web Conference 2021*, pages 798–808, 2021.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer et Richard Harshman : Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407, 1990.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp et Sebastian Riedel : Convolutional 2d knowledge graph embeddings. *In Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Jacob Devlin, Ming-Wei Chang, Kenton Lee et Kristina Toutanova : BERT : Pre-training of deep bidirectional transformers for language understanding. *In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and*

Short Papers), pages 4171–4186, Minneapolis, Minnesota, juin 2019. Association for Computational Linguistics. URL https://aclanthology.org/N19-1423.

Pedro Domingos : Bayesian averaging of classifiers and the overfitting problem. *In ICML*, volume 747, pages 223–230. Citeseer, 2000.

Xin Luna Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Kevin Murphy, Shaohua Sun et Wei Zhang : From data fusion to knowledge fusion. *arXiv preprint arXiv :1503.00302*, 2015.

Takuma Ebisu et Ryutaro Ichise : Toruse : Knowledge graph embedding on a lie group. *In Thirty-second AAAI conference on artificial intelligence*, 2018.

Joe Ellis, Jeremy Getman, Dana Fore, Neil Kuster, Zhiyi Song, Ann Bies et Stephanie M. Strassel : Overview of linguistic resources for the TAC KBP 2015 evaluations : Methodologies and results. *In Proceedings of the 2015 Text Analysis Conference, TAC 2015, Gaithersburg, Maryland, USA, November 16-17, 2015, 2015*. NIST, 2015. URL https://tac.nist.gov/publications/2015/additional.papers/TAC2015.KBP_resources_overview.proceedings.pdf.

Hady Elsahar, Pavlos Vougiouklis, Arslen Remaci, Christophe Gravier, Jonathon Hare, Frederique Laforest et Elena Simperl : T-REx : A large scale alignment of natural language with knowledge base triples. *In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, mai 2018. European Language Resources Association (ELRA). URL https://aclanthology.org/L18-1544.

John R Firth : A synopsis of linguistic theory, 1930-1955. *Studies in linguistic analysis*, 1957.

Robert M French : Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4):128–135, 1999.

Hao Fu, Shaojun Zhou, Qihong Yang, Junjie Tang, Guiquan Liu, Kaikui Liu et Xiaolong Li : Lrc-bert : latent-representation contrastive knowledge distillation for natural language understanding. *In Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12830–12838, 2021.

Jianping Gou, Baosheng Yu, Stephen J Maybank et Dacheng Tao : Knowledge distillation : A survey. *International Journal of Computer Vision*, 129(6):1789–1819, 2021.

Alex Graves, Abdel-rahman Mohamed et Geoffrey Hinton : Speech recognition with deep recurrent neural networks. *In 2013 IEEE international conference on acoustics, speech and signal processing*, pages 6645–6649. Ieee, 2013.

Maarten GROOTENDORST : Bertopic : Leveraging bert and c-tf-idf to create easily interpretable topics., 2020. URL https://doi.org/10.5281/zenodo.4381785.

Lingbing GUO, Zequn SUN et Wei HU : Learning to exploit long-term relational dependencies in knowledge graphs. *In International Conference on Machine Learning*, pages 2505–2514. PMLR, 2019.

Qiushan GUO, Xinjiang WANG, Yichao WU, Zhipeng YU, Ding LIANG, Xiaolin HU et Ping LUO : Online knowledge distillation via collaborative learning. *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Claudio GUTIERREZ et Juan F SEQUEDA : Knowledge graphs. *Communications of the ACM*, 64(3):96–104, 2021.

Kelvin GUU, Kenton LEE, Zora TUNG, Panupong PASUPAT et Mingwei CHANG : Retrieval augmented language model pre-training. *In* Hal Daumé III et Aarti SINGH, éditeurs : *Proceedings of the 37th International Conference on Machine Learning*, volume 119 de *Proceedings of Machine Learning Research*, pages 3929–3938. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/guu20a.html.

Xu HAN, Shulin CAO, Xin LV, Yankai LIN, Zhiyuan LIU, Maosong SUN et Juanzi LI : OpenKE : An open toolkit for knowledge embedding. *In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing : System Demonstrations*, pages 139–144, Brussels, Belgium, novembre 2018. Association for Computational Linguistics. URL https://aclanthology.org/D18-2024.

Bin HE, Di ZHOU, Jinghui XIAO, Xin JIANG, Qun LIU, Nicholas Jing YUAN et Tong XU : BERT-MK : Integrating graph contextualized knowledge into pre-trained language models. *In Findings of the Association for Computational Linguistics : EMNLP 2020*, pages 2281–2290, Online, novembre 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.findings-emnlp.207.

Kaiming HE, Xiangyu ZHANG, Shaoqing REN et Jian SUN : Deep residual learning for image recognition. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Martin HILBERT et Priscila LÓPEZ : The world's technological capacity to store, communicate, and compute information. *science*, 332(6025):60–65, 2011.

Geoffrey HINTON, Oriol VINYALS, Jeff DEAN *et al.* : Distilling the knowledge in a neural network. *arXiv preprint arXiv :1503.02531*, 2(7), 2015a.

Geoffrey Hinton, Oriol Vinyals et Jeffrey Dean : Distilling the knowledge in a neural network. *In NIPS Deep Learning and Representation Learning Workshop*, 2015b. URL http://arxiv.org/abs/1503.02531.

Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutierrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier *et al.* : Knowledge graphs. *ACM Computing Surveys (CSUR)*, 54(4):1–37, 2021.

Julia Holze : Dbpedia snapshot 2022-03 release, Jun 2022. URL https://www.dbpedia.org/blog/dbpedia-snapshot-2022-03-release/.

Ali Ismayilov, Dimitris Kontokostas, Sören Auer, Jens Lehmann, Sebastian Hellmann *et al.* : Wikidata through the eyes of dbpedia. *Semantic Web*, 9(4):493–503, 2018.

Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu et Jun Zhao : Knowledge graph embedding via dynamic mapping matrix. *In Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing (volume 1 : Long papers)*, pages 687–696, 2015.

Xiaotian Jiang, Quan Wang et Bin Wang : Adaptive convolution for multi-relational learning. *In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 978–987, Minneapolis, Minnesota, juin 2019. Association for Computational Linguistics. URL https://aclanthology.org/N19-1103.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang et Qun Liu : TinyBERT : Distilling BERT for natural language understanding. *In Findings of the Association for Computational Linguistics : EMNLP 2020*, pages 4163–4174, Online, novembre 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.findings-emnlp.372.

Jeff Johnson, Matthijs Douze et Hervé Jégou : Billion-scale similarity search with gpus. *arXiv preprint arXiv :1702.08734*, 2017.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer et Omer Levy : SpanBERT : Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77, 2020. URL https://aclanthology.org/2020.tacl-1.5.

Seyed Mehran Kazemi et David Poole : Simple embedding for link prediction in knowledge graphs. *Advances in neural information processing systems*, 31, 2018.

Pei Ke, Haozhe Ji, Siyang Liu, Xiaoyan Zhu et Minlie Huang : SentiLARE : Sentiment-aware language representation learning with linguistic knowledge. *In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6975–6988, Online, novembre 2020. Association for Computational Linguistics.

Yoon Kim et Alexander M Rush : Sequence-level knowledge distillation. *arXiv preprint arXiv :1606.07947*, 2016.

Quoc Le et Tomas Mikolov : Distributed representations of sentences and documents. *In International conference on machine learning*, pages 1188–1196. PMLR, 2014.

Jinhyuk Lee, Mujeen Sung, Jaewoo Kang et Danqi Chen : Learning dense representations of phrases at scale. *In Association for Computational Linguistics (ACL)*, 2021.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer *et al.* : Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic web*, 6(2):167–195, 2015.

Alessandro Lenci : Distributional models of word meaning. *Annual review of Linguistics*, 4:151–171, 2018.

Yoav Levine, Barak Lenz, Or Dagan, Ori Ram, Dan Padnos, Or Sharir, Shai Shalev-Shwartz, Amnon Shashua et Yoav Shoham : SenseBERT : Driving some sense into BERT. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4656–4667, Online, juillet 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.acl-main.423.

Omer Levy, Minjoon Seo, Eunsol Choi et Luke Zettlemoyer : Zero-shot relation extraction via reading comprehension. *In Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 333–342, Vancouver, Canada, août 2017. Association for Computational Linguistics. URL https://aclanthology.org/K17-1034.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov et Luke Zettlemoyer : BART : Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, juillet 2020a. Association for Computational Linguistics. URL https://aclanthology.org/2020.acl-main.703.

Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel et Douwe Kiela : Retrieval-augmented generation for knowledge-intensive nlp tasks. *In NeurIPS*, 2020b. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu et Xuan Zhu : Learning entity and relation embeddings for knowledge graph completion. *In Twenty-ninth AAAI conference on artificial intelligence*, 2015a.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu et Xuan Zhu : Learning entity and relation embeddings for knowledge graph completion. *In Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, page 2181–2187. AAAI Press, 2015b. ISBN 0262511290.

Hanxiao Liu, Yuexin Wu et Yiming Yang : Analogical inference for multi-relational embeddings. *In International conference on machine learning*, pages 2168–2178. PMLR, 2017.

Quan Liu, Hui Jiang, Zhen-Hua Ling, Si Wei et Yu Hu : Probabilistic reasoning via deep learning : Neural association models. *CoRR*, abs/1603.07704, 2016. URL http://arxiv.org/abs/1603.07704.

Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Haotang Deng et Qi Ju : Fastbert : a self-distilling bert with adaptive inference time. *arXiv preprint arXiv :2004.02178*, 2020.

Yijia Liu, Wanxiang Che, Huaipeng Zhao, Bing Qin et Ting Liu : Distilling knowledge for search-based structured prediction. *In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 1393–1402, 2018.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer et Veselin Stoyanov : Roberta : A robustly optimized bert pretraining approach. *arXiv preprint arXiv :1907.11692*, 2019.

Yi Luan, Luheng He, Mari Ostendorf et Hannaneh Hajishirzi : Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. *In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3219–3232, Brussels, Belgium, octobre-novembre 2018. Association for Computational Linguistics. URL https://aclanthology.org/D18-1360.

Kevin Lund et Curt Burgess : Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior research methods, instruments, & computers*, 28(2): 203–208, 1996.

Christopher MANNING et Hinrich SCHUTZE : *Foundations of statistical natural language processing*. MIT press, 1999.

Tomas MIKOLOV, Kai CHEN, Greg CORRADO et Jeffrey DEAN : Efficient estimation of word representations in vector space. *arXiv preprint arXiv :1301.3781*, 2013a.

Tomas MIKOLOV, Ilya SUTSKEVER, Kai CHEN, Greg S CORRADO et Jeff DEAN : Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013b.

George A MILLER et Walter G CHARLES : Contextual correlates of semantic similarity. *Language and cognitive processes*, 6(1):1–28, 1991.

Pasquale MINERVINI, Claudia D'AMATO, Nicola FANIZZI et Floriana ESPOSITO : Efficient learning of entity and predicate embeddings for link prediction in knowledge graphs. *URSW@ ISWC*, 1479:26–37, 2015.

Dai Quoc NGUYEN, Tu Dinh NGUYEN, Dat Quoc NGUYEN et Dinh PHUNG : A novel embedding model for knowledge base completion based on convolutional neural network. *In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 2 (Short Papers)*, pages 327–333, New Orleans, Louisiana, juin 2018. Association for Computational Linguistics. URL https://aclanthology.org/N18-2053.

Dai Quoc NGUYEN, Thanh VU, Tu Dinh NGUYEN, Dat Quoc NGUYEN et Dinh PHUNG : A capsule network-based embedding model for knowledge graph completion and search personalization. *In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2180–2189, Minneapolis, Minnesota, juin 2019. Association for Computational Linguistics. URL https://aclanthology.org/N19-1226.

Dat Quoc NGUYEN, Kairit SIRTS, Lizhen QU et Mark JOHNSON : STransE : a novel embedding model of entities and relationships in knowledge bases. *In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, pages 460–466, San Diego, California, juin 2016. Association for Computational Linguistics. URL https://aclanthology.org/N16-1054.

Maximilian NICKEL, Lorenzo ROSASCO et Tomaso POGGIO : Holographic embeddings of knowledge graphs. *In Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016.

Maximilian NICKEL, Volker TRESP et Hans-Peter KRIEGEL : A three-way model for collective learning on multi-relational data. *In Icml*, 2011.

Peter Oram : Wordnet : An electronic lexical database. christiane fellbaum (ed.). cambridge, ma : Mit press, 1998. pp. 423. *Applied Psycholinguistics*, 22(1):131–134, 2001.

Thomas Pellissier Tanon, Denny Vrandečić, Sebastian Schaffert, Thomas Steiner et Lydia Pintscher : From freebase to wikidata : The great migration. *In Proceedings of the 25th international conference on world wide web*, pages 1419–1428, 2016.

Jeffrey Pennington, Richard Socher et Christopher D Manning : Glove : Global vectors for word representation. *In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee et Luke Zettlemoyer : Deep contextualized word representations. *In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, juin 2018. Association for Computational Linguistics. URL https://aclanthology.org/N18-1202.

Matthew E. Peters, Mark Neumann, Robert L Logan, Roy Schwartz, Vidur Joshi, Sameer Singh et Noah A. Smith : Knowledge enhanced contextual word representations. *In EMNLP*, 2019.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, Vassilis Plachouras, Tim Rocktäschel et Sebastian Riedel : KILT : a benchmark for knowledge intensive language tasks. *In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, pages 2523–2544, Online, juin 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.naacl-main.200.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu et Alexander Miller : Language models as knowledge bases ? *In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China, novembre 2019. Association for Computational Linguistics. URL https://aclanthology.org/D19-1250.

Pouya Pezeshkpour, Liyan Chen et Sameer Singh : Embedding multimodal relational data for knowledge base completion. *In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3208–3218, Brussels, Belgium, octobre-novembre 2018. Association for Computational Linguistics. URL https://aclanthology.org/D18-1359.

Nina POERNER, Ulli WALTINGER et Hinrich SCHÜTZE : E-BERT : Efficient-yet-effective entity embeddings for BERT. *In Findings of the Association for Computational Linguistics : EMNLP 2020*, pages 803–818, Online, novembre 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.findings-emnlp.71.

Xipeng QIU, Tianxiang SUN, Yige XU, Yunfan SHAO, Ning DAI et Xuanjing HUANG : Pre-trained models for natural language processing : A survey. *Science China Technological Sciences*, 63(10):1872–1897, 2020.

Alec RADFORD, Karthik NARASIMHAN, Tim SALIMANS, Ilya SUTSKEVER *et al.* : Improving language understanding by generative pre-training. 2018.

Alec RADFORD, Jeffrey WU, Rewon CHILD, David LUAN, Dario AMODEI, Ilya SUTSKEVER *et al.* : Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Nils REIMERS et Iryna GUREVYCH : Sentence-bert : Sentence embeddings using siamese bert-networks. *In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL https://arxiv.org/abs/1908.10084.

Haopeng REN, Yi CAI, Xiaofeng CHEN, Guohua WANG et Qing LI : A two-phase prototypical network model for incremental few-shot relation classification. *In Proceedings of the 28th International Conference on Computational Linguistics*, pages 1618–1629, Barcelona, Spain (Online), décembre 2020. International Committee on Computational Linguistics. URL https://aclanthology.org/2020.coling-main.142.

Adriana ROMERO, Nicolas BALLAS, Samira Ebrahimi KAHOU, Antoine CHASSANG, Carlo GATTA et Yoshua BENGIO : Fitnets : Hints for thin deep nets. *International Conference on Learning Representations*, 2015.

Andrea ROSSI, Denilson BARBOSA, Donatella FIRMANI, Antonio MATINATA et Paolo MERIALDO : Knowledge graph embedding for link prediction : A comparative analysis. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 15(2):1–49, 2021.

Victor SANH, Lysandre DEBUT, Julien CHAUMOND et Thomas WOLF : Distilbert, a distilled version of BERT : smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108, 2019. URL http://arxiv.org/abs/1910.01108.

Bharat Bhusan SAU et Vineeth N. BALASUBRAMANIAN : Deep model compression : Distilling knowledge from noisy teachers. *CoRR*, abs/1610.09650, 2016. URL http://arxiv.org/abs/1610.09650.

Hinrich Schütze : Automatic word sense discrimination. *Computational Linguistics*, 24(1):97–123, 1998. URL https://aclanthology.org/J98-1004.

Amit Singhal : Introducing the knowledge graph : things, not strings, 2012. URL https://www.blog.google/products/search/introducing-knowledge-graph-things-not/.

Richard Socher, Danqi Chen, Christopher D Manning et Andrew Ng : Reasoning with neural tensor networks for knowledge base completion. *Advances in neural information processing systems*, 26, 2013.

Raphaël Sourty, Jose G. Moreno, François-Paul Servant et Lynda Tamine-Lechani : Knowledge base embedding by cooperative knowledge distillation. *In Proceedings of the 28th International Conference on Computational Linguistics*, pages 5579–5590, Barcelona, Spain (Online), décembre 2020. International Committee on Computational Linguistics. URL https://www.aclweb.org/anthology/2020.coling-main.489.

Raphael Sourty, Jose G. Moreno, Lynda Tamine et Francois-Paul Servant : Cherche : A new tool to rapidly implement pipelines in information retrieval. *In Proceedings of SIGIR 2022*, 2022a.

Raphael Sourty, Jose G. Moreno, Lynda Tamine et Francois-Paul Servant : Using cherche to empower newcomers into neural information retrieval. *In Proceedings of CIRCLE 2022*, 2022b.

Samuel Stanton, Pavel Izmailov, Polina Kirichenko, Alexander A Alemi et Andrew G Wilson : Does knowledge distillation really work ? *Advances in Neural Information Processing Systems*, 34, 2021.

Siqi Sun, Yu Cheng, Zhe Gan et Jingjing Liu : Patient knowledge distillation for bert model compression. *arXiv preprint arXiv :1908.09355*, 2019a.

Tianxiang Sun, Yunfan Shao, Xipeng Qiu, Qipeng Guo, Yaru Hu, Xuanjing Huang et Zheng Zhang : CoLAKE : Contextualized language and knowledge embedding. *In Proceedings of the 28th International Conference on Computational Linguistics*, pages 3660–3670, Barcelona, Spain (Online), décembre 2020a. International Committee on Computational Linguistics. URL https://aclanthology.org/2020.coling-main.327.

Zequn Sun, Wei Hu, Qingheng Zhang et Yuzhong Qu : Bootstrapping entity alignment with knowledge graph embedding. *In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 4396–4402. International Joint Conferences on Artificial Intelligence Organization, 7 2018. URL https://doi.org/10.24963/ijcai.2018/611.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie et Jian Tang : Rotate : Knowledge graph embedding by relational rotation in complex space. *In International Conference on Learning Representations*, 2019b. URL https://openreview.net/forum?id=HkgEQnRqYQ.

Zhiqing Sun, Shikhar Vashishth, Soumya Sanyal, Partha Talukdar et Yiming Yang : A re-evaluation of knowledge graph completion methods. *In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5516–5522, Online, juillet 2020b. Association for Computational Linguistics. URL https://aclanthology.org/2020.acl-main.489.

Mihai Surdeanu et Heng Ji : Overview of the english slot filling track at the tac 2014 knowledge base population evaluation. 2014.

Jizhi Tang, Yansong Feng et Dongyan Zhao : Learning to update knowledge graphs by reading news. *In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2632–2641, Hong Kong, China, novembre 2019a. Association for Computational Linguistics. URL https://aclanthology.org/D19-1265.

Raphael Tang, Yao Lu, Linqing Liu, Lili Mou, Olga Vechtomova et Jimmy Lin : Distilling task-specific knowledge from bert into simple neural networks. *arXiv preprint arXiv :1903.12136*, 2019b.

Kristina Toutanova et Danqi Chen : Observed versus latent features for knowledge base and text inference. *In Proceedings of the 3rd workshop on continuous vector space models and their compositionality*, pages 57–66, 2015.

Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoifung Poon, Pallavi Choudhury et Michael Gamon : Representing text for joint embedding of text and knowledge bases. *In Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1499–1509, 2015.

Bayu Distiawan Trisedya, Jianzhong Qi et Rui Zhang : Entity alignment between knowledge graphs using attribute embeddings. *In Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 297–304, 2019.

Rakshit Trivedi, Bunyamin Sisman, Xin Luna Dong, Christos Faloutsos, Jun Ma et Hongyuan Zha : LinkNBed : Multi-graph representation learning with entity linkage. *In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 252–262, Melbourne, Australia, juillet 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P18-1024.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier et Guillaume Bouchard : Complex embeddings for simple link prediction. International Conference on Machine Learning (ICML), 2016.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser et Illia Polosukhin : Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Denny Vrandečić et Markus Krötzsch : Wikidata : a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014.

Quan Wang, Zhendong Mao, Bin Wang et Li Guo : Knowledge Graph Embedding : A Survey of Approaches and Applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12):2724–2743, décembre 2017. ISSN 1041-4347.

Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Guihong Cao, Daxin Jiang, Ming Zhou *et al.* : K-adapter : Infusing knowledge into pre-trained models with adapters. *arXiv preprint arXiv :2002.01808*, 2020a.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang et Ming Zhou : Minilm : Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems*, 33:5776–5788, 2020b.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li et Jian Tang : KEPLER : A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 9:176–194, 2021a. URL https://aclanthology.org/2021.tacl-1.11.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li et Jian Tang : Kepler : A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 9:176–194, 2021b.

Zhen Wang, Jianwen Zhang, Jianlin Feng et Zheng Chen : Knowledge graph embedding by translating on hyperplanes. *In Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, page 1112–1119. AAAI Press, 2014a.

Zhen Wang, Jianwen Zhang, Jianlin Feng et Zheng Chen : Knowledge graph embedding by translating on hyperplanes. *In Proceedings of the AAAI conference on artificial intelligence*, volume 28, 2014b.

Zhen Wang, Jianwen Zhang, Jianlin Feng et Zheng Chen : Knowledge graph embedding by translating on hyperplanes. *In Proceedings of the Twenty-Eighth*

AAAI Conference on Artificial Intelligence, AAAI'14, page 1112–1119. AAAI Press, 2014c.

Liu Weijie, Zhou Peng, Zhao Zhe, Wang Zhiruo, Ju Qi, Deng Haotang et Wang Ping : K-BERT : Enabling language representation with knowledge graph. *In Proceedings of AAAI 2020*, 2020.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest et Alexander Rush : 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, Online, octobre 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.emnlp-demos.6.

David H Wolpert : Stacked generalization. *Neural networks*, 5(2):241–259, 1992.

Chuhan Wu, Fangzhao Wu et Yongfeng Huang : One teacher is enough ? pre-trained language model distillation from multiple teachers. *arXiv preprint arXiv :2106.01023*, 2021.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda et Yuji Matsumoto : LUKE : Deep contextualized entity representations with entity-aware self-attention. *In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6442–6454, Online, novembre 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.emnlp-main.523.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao et Li Deng : Embedding entities and relations for learning and inference in knowledge bases. *In* Yoshua Bengio et Yann LeCun, éditeurs : *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015a. URL http://arxiv.org/abs/1412.6575.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao et Li Deng : Embedding entities and relations for learning and inference in knowledge bases. *In* Yoshua Bengio et Yann LeCun, éditeurs : *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015b. URL http://arxiv.org/abs/1412.6575.

Jian Yang, Gang Xiao, Yulong Shen, Wei Jiang, Xinyu Hu, Ying Zhang et Jinghui Peng : A survey of knowledge enhanced pre-trained models. *arXiv preprint arXiv :2110.00269*, 2021.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov et Quoc V Le : Xlnet : Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2019.

Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao et Kaisheng Ma : Be your own teacher : Improve the performance of convolutional neural networks via self distillation. *In Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3713–3722, 2019a.

Qingheng Zhang, Zequn Sun, Wei Hu, Muhao Chen, Lingbing Guo et Yuzhong Qu : Multi-view knowledge graph embedding for entity alignment. *In Proceedings of the 28th International Joint Conference on Artificial Intelligence*, IJCAI'19, pages 5429–5435, 08 2019b.

Wen Zhang, Bibek Paudel, Wei Zhang, Abraham Bernstein et Huajun Chen : Interaction embeddings for prediction and explanation in knowledge graphs. *In Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pages 96–104, 2019c.

Y. Zhang, T. Xiang, T. M. Hospedales et H. Lu : Deep mutual learning. *In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4320–4328, 2018.

Ying Zhang, Tao Xiang, Timothy M Hospedales et Huchuan Lu : Deep mutual learning. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4320–4328, 2018.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun et Qun Liu : ERNIE : Enhanced language representation with informative entities. *In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy, juillet 2019d. Association for Computational Linguistics. URL https://aclanthology.org/P19-1139.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun et Qun Liu : ERNIE : Enhanced language representation with informative entities. *In Proceedings of ACL 2019*, 2019e.

Jiawei Zhao, Wei Luo, Boxing Chen et Andrew Gilman : Mutual-learning improves end-to-end speech translation. *In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3989–3994, Online and Punta Cana, Dominican Republic, novembre 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.emnlp-main.325.

Xiaojuan Zhao, Yan Jia, Aiping Li, Rong Jiang et Yichen Song : Multi-source knowledge fusion : a survey. *World Wide Web*, 23(4):2567–2592, 2020.

Hao ZHU, Ruobing XIE, Zhiyuan LIU et Maosong SUN : Iterative entity alignment via joint knowledge embeddings. *In Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, page 4258–4264. AAAI Press, 2017a. ISBN 9780999241103.

Hao ZHU, Ruobing XIE, Zhiyuan LIU et Maosong SUN : Iterative entity alignment via knowledge embeddings. *In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2017b.

Barret ZOPH, Golnaz GHIASI, Tsung-Yi LIN, Yin CUI, Hanxiao LIU, Ekin D. CUBUK et Quoc V. LE : Rethinking pre-training and self-training, 2020.