Reconnaissance d'inférences textuelles en présence de marqueurs discursifs, à l'aide de méthodes hybrides

Présentation du stage recherche

Master *Sciences et Technologies*,
Mention *Informatique*,
Parcours CMI IASD

Auteur
HADDAD Gatien

Superviseurs
PONCELET Pascal (LIRMM)
RETORÉ Christian (LIRMM)
WINTERSTEIN Grégoire (UQÀM)

Lieu de stage
LIRMM UM5506 - CNRS, Université de Montpellier, France
Pavillon Hubert-Aquin, A-3405 400, rue Sainte-Catherine Est, UQÀM, Canada

14 mars 2024

Résumé

Les avancées en Machine Learning ont révolutionné le traitement automatique des langues, mais les modèles actuels peinent dans les tâches impliquant le raisonnement logique, comme la reconnaissance d'inférences textuelles. Notre travail vise à explorer une méthode hybride combinant l'apprentissage automatique et les règles linguistiques pour traiter les inférences textuelles, en se concentrant sur les opérateurs discursifs tels que "presque" et "à peine". Nous analysons ces opérateurs à l'aide de la théorie de l'argumentation linguistique et construisons un corpus annoté pour entraîner un système de prédiction des relations argumentatives. Notre approche vise à améliorer les modèles basés uniquement sur le Machine Learning en intégrant des connaissances linguistiques pour une meilleure compréhension des inférences textuelles.

Abstract

Advancements in machine learning have revolutionized natural language processing, but current models struggle with tasks involving logical reasoning, such as recognizing textual inferences. Our work aims to explore a hybrid method combining machine learning and linguistic rules to address textual inferences, focusing on French discourse operators such as "presque" (almost) and "à peine" (barely). We analyze these operators using linguistic argumentation theory and construct an annotated corpus to train a system for predicting argumentative relations. Our approach aims to enhance models solely based on machine learning by integrating linguistic knowledge for a better understanding of textual inferences.

Table des matières

1. Introduction

Dans un monde où l'utilisation des technologies du langage naturel est devenue omniprésente, la compréhension précise et profonde du langage reste cependant un défi majeur. Le Traitement Automatique du Langage Naturel (TALN) joue un rôle crucial dans ce domaine, en permettant aux ordinateurs de comprendre du texte en le transformant en représentations utilisables par une machine, et également pouvoir générer du langage de manière automatique. Au coeur de cette problématique se trouve la capacité à reconnaitre des inférences textuelles, c'est-à-dire pouvoir déduire des informations à partir des éléments d'un texte.

De plus, nous avons récemment pu constater que l'avènement des Grands Modèles de Langue (LLM en Anglais) a révolutionné le domaine du TALN en permettant des avancées significatives dans des tâches telles que la traduction automatique ou la génération de texte. Cependant, malgré ces progrès, les LLM sont encore confrontés à des difficultés lorsqu'il s'agit de comprendre les inférences textuelles et les nuances du langage. Les marqueurs discursifs, tels que "presque" et "à peine", sont un exemple de défis particuliers en raison de leur capacité à modifier subtilement le sens d'une phrase selon le contexte.

Ce stage se propose d'explorer des méthodes hybrides pour aborder les inférences textuelles en présence de marqueurs discursifs. Nous commencerons par définir la question étudiée, puis nous présenterons une étude bibliographique qui guidera notre résolution du problème, où nous présenterons nos méthodologies proposées et les objectifs visés dans le cadre de ce stage de recherche.

2. Qu'est-ce que le Traitement Automatique du Langage Naturel

2.1 Définition du TALN

2.1.1 Définition et intérêt du TALN

Le langage naturel désigne le mode de communication servant à s'exprimer à l'oral ou à l'écrit, en utilisant des règles propres à chaque langue. Ces règles peuvent être syntaxiques, lexicales et sémantiques, et témoignent de la diversité des expressions linguistiques, en allant des conversations entre amis aux discours formels, et permettent d'exprimer une large gamme de nuances.
Permettre aux machines de comprendre le langage naturel est essentiel pour permettre à ces dernières d'interagir de manière plus naturelle avec les utilisateurs. Par exemple, ces systèmes sont utilisés pour développer des *Chatbots* pour répondre aux questions des utilisateurs, des outils de traduction automatique, ou encore des outils d'analyse de texte pour extraire des informations pertinentes à partir de grandes quantités de données.
Le Traitement Automatique du Langage Naturel (TALN) à alors été introduit pour contribuer à la compréhension du langage naturel par les machines. Ce domaine de recherche se situe à cheval entre l'informatique et la linguistique, et vise à développer des algorithmes capables de traiter automatiquement et de manière intelligente le langage naturel. Son objectif principal est de permettre aux ordinateurs de comprendre, d'interpréter et d'engendrer du langage de la même manière que nous sommes capables de faire.

2.1.2 Méthodes utilisées

Les méthodes de TALN englobent une multitude de techniques qui visent à permettre aux machines de comprendre et de générer du texte de manière automatique. Parmi ces méthodes, il y a dans un premier temps les méthodes symboliques, où les règles linguistiques sont explicitement codées dans le programme. Cette approche implique souvent des tâches telles que l'encodage des règles grammaticales, la manipulation des structures syntaxiques et la définition des modèles sémantiques. En utilisant des méthodes symboliques, il est possible de construire des systèmes TALN qui fonctionnent selon des règles préétablies, imitant ainsi le processus de réflexion humain lors de la compréhension du langage naturel.

D'autre part, les méthodes d'apprentissage automatique (telles que les réseaux de neurones) représentent une approche plus axée sur les données, où les modèles apprennent à partir de grands ensembles de données textuelles. Ces modèles linguistiques

semblent capables de généraliser à partir des exemples fournis et peuvent être utilisés pour une grande variété de tâches, comme la classification de texte (détection de fake news, etc), la traduction ainsi que la génération de texte.

2.2 Importance des inférences textuelles dans le TALN

2.2.1 Définition et intérêt

Les inférences textuelles désignent la capacité à déduire des informations à partir d'un texte. Cette capacité va au-delà de la simple reconnaissance des mots, et requiert la compréhension du sens profond et des relations sémantiques entre les énoncés. Les inférences textuelles permettent de saisir les nuances et les ambiguïtés qui pourraient être présentes dans un texte, ce qui est essentiel pour une compréhension complète du langage. En effet, nous utilisons souvent des inférences textuelles pour tirer des conclusions, interpréter des intentions et anticiper des réponses lors de discussions quotidiennes. Par exemple, si l'on nous dit "Alice regarda le ciel, elle comprit qu'il allait pleuvoir", nous pouvons naturellement en déduire que Alice a vu des nuages.

Le processus est le même pour une machine à qui nous demanderions "Qui a peint La Joconde ?". Pour répondre correctement "Léonard de Vinci", notre modèle doit être capable de comprendre que cette réponse découle logiquement d'une information présente dans une autre phrase qui a été utilisée lors de l'entraînement. Par exemple, si nous lui donnons comme information préalable la phrase "Au Louvre est exposée La Joconde de Léonard de Vinci", notre machine doit être en mesure d'inférer la phrase "Léonard de Vinci a peint La Joconde" à partir du texte initial. En poussant la notion de déduction, nous pourrions dire que notre modèle doit être capable de comprendre que l'artiste associé à l'exposition de La Joconde au Louvre est également celui qui l'a peinte. C'est donc ce processus de déduction et de compréhension du sens profond des énoncés qui caractérise l'inférence textuelle.

Formellement, l'inférence textuelle est une relation dirigée d'un texte (ou prémisse) T vers une hypothèse H. Une hypothèse H est inférée par un texte T, si un humain qui lirait T déduirait que H est (très probablement) vraie. Comme nous pouvons le remarquer, cette définition semble un peu abstraite car elle se base sur "ce qu'une personne déduirait". Cependant, cela est contrebalancé grâce à la création de jeux de données de qualité supérieure, c'est à dire des données d'évaluation qui sont manuellement annotées selon des normes établies, et qui contiennent généralement un large éventail d'exemples couvrant différents cas d'utilisation du domaine étudié [1].
Les utilisations des inférences textuelles se concentrent sur la réponses à des question à partir de textes, comme par exemple au travers de la détection de sentiment dans des commentaires (inférer s'ils sont positifs, négatifs, etc), dans les moteurs de recherche pour extraire les informations pertinentes, ou encore dans les systèmes de résumés automatiques.

2.2.2 Approches traditionnelles des inférences textuelles

Les méthodes basées sur des règles de logique formelle sont depuis longtemps utilisées pour aborder les inférences textuelles en TALN. Il s'agit d'utiliser des règles logiques et linguistique, qui sont utilisées pour déduire de nouvelles informations à partir des informations déjà présentes dans le texte. Par exemple, pour la règle du modus ponens, si l'on a la phrase "A implique B" et la phrase "A", alors on peut conclure que la phrase "B" est vraie. Ces méthodes ont été largement utilisées pour modéliser des tâches telles que de l'analyse sémantique des énoncés ou de la déduction formelle dans des domaines spécifiques [1].
Ces méthodes se basent sur des algorithmes de traitement symbolique du langage, qui constituent donc une approche traditionnelle pour aborder les inférences textuelles. En combinant un grand nombre de règles, il est donc possible de construire des systèmes de TALN capables de réaliser une variété de tâches liées à l'analyse et à la compréhension du langage naturel.

2.2.3 Limitations des approches traditionnelles

Bien que les approches symboliques offrent une base solide pour le traitement automatique du langage naturel en raison de leur fondement dans la logique formelle, elles présentent également des limites. Ces méthodes reposent souvent sur des règles préétablies, ce qui les rend généralement fiables selon la précision de ces règles. Cependant, il est difficile de généraliser ces règles à toutes les situations possibles, car cela nécessiterait de prendre en compte un trop grand nombre de cas particuliers. Par conséquent, les approches symboliques peuvent être limitées dans leur capacité à traiter efficacement la grande variabilité du langage naturel, en particulier dans des domaines ou des contextes en constante évolution[2]. Cette difficulté à généraliser les règles symboliques constitue l'un des principaux défis auxquels sont confrontées les approches traditionnelles du traitement du langage naturel, et souligne la nécessité de développer des méthodes plus flexibles et adaptatives pour surmonter ces limitations.

2.2.4 Introduction aux méthodes de Machine Learning

Une des approches existante consiste alors à se tourner vers le Machine Learning (ML). Ce dernier suscite un intérêt croissant dans de nombreux domaines, ce qui nous intéresse particulièrement pour obtenir des solutions adaptatives pour traiter la complexité du langage naturel.

Le Machine Learning est une branche de l'intelligence artificielle qui se focalise sur la capacité des machines à apprendre, seules, à partir de données, sans nécessiter de programmation spécifique pour chaque tâche différente. Comme nous l'avons évoqué plus tôt, le ML trouve des applications dans une multitude de domaines, allant de la traduction automatique à la génération de texte, en passant par l'analyse sentimentale sur les réseaux sociaux.

---

[1] Jean-Claude Anscombre and Oswald Ducrot - *L'argumentation dans la langue* [2]

[2] Comme par exemple des données médicales, ou des discutions sur les réseaux sociaux

Pour réaliser ce genre de tâche pour du TALN, il est essentiel d'avoir un corpus de données qui servira de base d'entraînement pour le modèle d'inférences textuelles. Ce corpus doit contenir une variété d'exemples de texte, accompagnés d'annotations détaillées sur les relations sémantiques et logiques entre les phrases. Ces annotations peuvent inclure des informations sur les relations d'inférences, de contradiction ou de neutralité entre les énoncés.
Une fois le corpus annoté créé, différents modèles d'inférences textuelles peuvent être entraînés sur ce corpus, puis doivent être évalués sur un ensemble de données de test pour évaluer leurs performances et leur capacité à généraliser à de nouveaux exemples de texte. Ce processus d'entraînement et d'évaluation permet d'améliorer progressivement les performances des modèles d'inférences textuelles et de les adapter à des tâches spécifiques.

Ainsi, en exploitant de grandes quantités de données et en utilisant des algorithmes qui visent à imiter le cerveau humain tels que les réseaux de neurones, le ML permet de construire des systèmes de TALN capables d'apprendre à partir d'exemples et de s'adapter à des situations variées.
Cependant, il est assez contraignant de devoir s'entraîner sur autant de données, d'une part car cela requiert d'avoir de grandes puissances de calcul pour pouvoir entraîner puis faire tourner ces modèles, et d'autre part car selon la tâche nécessaire, un corpus de cette envergure n'est pas toujours disponible et il faut alors pouvoir extraire ces données en grande quantité, ce qui nécessite également beaucoup de temps.

2.2.5 Approche hybride

C'est dans un but de tirer parti des avantages respectifs de chaque méthode pour améliorer les performances des systèmes de TALN qu'est apparue l'approche par des méthodes hybrides. En combinant les méthodes traditionnelles basées sur des règles symboliques avec les techniques d'apprentissage autonome du ML, il est possible de construire des systèmes hybrides capables de bénéficier à la fois de la précision des règles symboliques, ainsi que la capacité d'adaptation et l'efficacité du ML.

La combinaison des méthodes traditionnelles et du ML peut se faire de plusieurs manières différentes. Par exemple, les règles symboliques peuvent être utilisées pour guider l'apprentissage des modèles de ML en fournissant des informations structurées sur la langue pour implémenter le processus d'inférence. D'un autre côté, les modèles de ML peuvent être utilisés pour apprendre à partir de grandes quantités de données et être capables de généraliser sur de nouveaux exemples, ce qui peut compenser les limitations des approches traditionnelles en termes d'adaptabilité.

C'est en ce sens que l'objectif de ce travail est de mettre en place une méthode hybride permettant de traiter efficacement les inférences textuelles. De plus, le but est de ce concentrer sur des opérateurs argumentatifs comme “presque” et “à peine”, qui permettent d'exprimer des enchaînements de phrases qui sont difficiles à exprimer en se basant uniquement sur leur signification de base.

Par exemple pour l'opérateur "presque", bien qu'il signifie d'un point de vue logique qu'une chose n'est **pas** terminée, il existe des cas où sa présence vient argumenter le contraire. Considérons l'exemple suivant :
(a) J'ai *presque* fini ma bière, commande-moi en une autre.
(b) # Je n'ai pas fini ma bière, commande-moi en une autre.
D'un point de vue logique, l'adverbe *presque* en (a) exprime le fait que la bière n'est **pas** terminée. Cependant, l'exemple (a) apparaît naturel, contrairement à (b) qui exprime pourtant la même information.

Ainsi, le but de ce stage est de développer un système hybride de reconnaissance d'inférences textuelles qui combine les avantages du ML et des méthodes symboliques pour obtenir des performances optimales dans une variété de tâches liées au traitement automatique du langage naturel.
La première étape consistera à créer un corpus de phrases contenant des opérateurs argumentatifs comme vu plus haut. Ce corpus contiendra des annotations entre chaque phrase et son préjacent, dans le cas vu plus haut, nous aurions alors :

- *Énoncé :* Il est presque mort, l'ambulance doit accélérer.
- *Préjacent :* Il est mort, l'ambulance doit accélérer.
- *Similarité argumentative :* Opposition
- *Relation d'implication :* Négation

Une des approches d'extraction, que nous reverrons plus en détail dans la section 4.2, consisterait à extraire des phrases contenant des adverbes spécifiques depuis de grosses banques de textes (Wikipedia, Common Crawl, etc), puis à les annoter semi-automatiquement pour identifier les relations d'inférence entre les paires de phrases.

Le but de ce corpus est ensuite de pouvoir mettre en place un système qui viendrait prédire les relations entre les phrases (similarité argumentative et relation d'implication). Par la suite, le but est d'enrichir le système en intégrant une approche symbolique à l'aide de règles pour essayer d'améliorer la qualité des prédictions.

3. État de l'Art

Dans cette section, nous allons explorer les approches existantes pour traiter les marqueurs discursifs dans le domaine du TALN. Ces marqueurs discursifs, tels que "presque" et "à peine" dans ce rapport, jouent un rôle crucial dans l'analyse du langage naturel en introduisant des nuances dans le discours. Depuis plusieurs décennies, des études linguistiques et d'informatique symbolique ont été menées pour étudier le fonctionnement de ces marqueurs et développer des méthodes efficaces pour les traiter automatiquement.

3.1 Approches existantes pour traiter les marqueurs discursifs

Une partie de la recherche dans le domaine de la linguistique se focalise sur la manière dont les informations peuvent être inférées de manière (plus ou moins) fiable à partir d'un texte, ainsi que sur le raisonnement basé sur ces informations, que ce soit en utilisant des règles linguistiques, ou bien des inférences statistiques basés sur les caractéristiques des textes représentés [1]. Pour évaluer les performances de tels modèles de langues, il existe notamment des bechmarks qui fournissent une batterie de tâches de compréhension du langage naturel, tells que GLUE ou SuperGLUE [3, 4], qui offrent une plusieurs outils pour évaluer les performances des modèles sur un ensemble diversifié de tâches en TALN, et permettre de mesurer leur robustesse et leur capacité d'adaptation.

Un importante partie de la recherche s'effectue également autour du Machine Learning. En effet et comme présenté dans l'introduction, les Grands Modèles de Langues ont connu une croissance importante ces dernières années, et ont donc été utilisés pour des tâches de reconnaisances d'inférences textuelles [2].

3.2 Lacunes et défis à relever

Bien que ces modèles offrent une méthode pour traiter les inférences textuelles, les LLM ont cependant une propension naturelle à générer des hallucinations [3]. Les hallucinations sont des informations qui apparaissent lorsque les modèles génèrent des informations inexactes voire incorrectes, souvent en raison de biais dans les données d'entraînement ou dans les algorithmes d'apprentissage. Ces hallucinations peuvent donc compromettre la fiabilité et la pertinence des résultats produits pour des systèmes d'inférences textuelles.

[1] Ido Dagan et al. - *Recognizing Textual entailment : models and applications* [1]
[2] Ido Dagan, Dan Roth, Mark Sammons, and Fabio Zanzotto - *Recognizing textual entailment* [1]
[3] Yue Zhang et al. - *A survey on hallucination in large language models* [5]

De plus, les LLM ont également du mal à prendre en compte les nuances sémantiques d'une variété d'opérateurs présents dans le langage naturel. Un des exemple est notamment l'opérateur d'inversion [4]. Lorsqu'un modèle est entraîné avec une phrase comme "Leonard De Vinci peint La Joconde", il ne sera pas nécessairement en mesure de répondre à la question inverse, à savoir "Qui à peint La Joconde ?"

Dans le cas des inférences textuelles, il a été montré que de nombreux jeux de données contenaient des artefacts d'annotations, c'est à dire des biais introduits par des humains lors du processus d'annotation des données, ce qui peut fausser les résultats des modèles. En effet, les modèles entraînés sur ces corpus arrivaient à prédire la relation d'inférence entre le texte et l'hypothèse sans même lire le texte, notamment grâce à des contradictions dans les hypothèses. Par exemple, l'hypothèse "Thomas n'était pas présent au partiel" était considérée par le modèle comme une contradiction avec le texte "Thomas s'est rendu au partiel", sans même connaître ce dernier, en raison de la négation dans l'hypothèse [5].

Il arrive toutefois que certains LLM, entraînés avec de grandes quantités de données performent tout de même bien sur les tâches d'inférences. C'est par exemple le cas de ChatGPT-3.5 [8], avec ses 175 milliards de paramètres, qui à première vue peut sembler être capable de capturer les nuances nécessaires pour effectuer des inférences textuelles. Toutefois, il est important de noter que ces même modèles sont également influencés par les biais qui ont été évoqués précédemment. Ces biais peuvent se manifester à travers des nuances telles que la négation et des imprécisions, qui peuvent conduire à des erreurs d'inférence et compromettre la performance globale du modèle [6]. Ainsi, bien que certains LLM puissent obtenir des résultats prometteurs sur les inférences, ces résultats n'en restent pas moins sur-estimés, et il reste donc crucial d'essayer de repousser les limites dans le domaine des inférences textuelles.

En outre, le résultat de l'entrainement massif de ces modèles encode des informations argumentatives, mais en un sens, ces informations sont dénaturées lorsqu'on demande a ces modèles de faire de l'inférence logique. Ces derniers ayant appris à reconnaitre des similarités distributionnelles, cela peut être comparé à de la similarité argumentative : si on peut remplacer un énoncé par un autre sans affecter la coherence du discours, peu importe que l'un implique la négation de l'autre, du moment que le discours reste naturel et coherent [7].

Pour illustrer cela, considérons un modèle qui a été entraîné sur un large corpus de texte où il a rencontré des phrases comme "La Joconde a été peinte par Léonard de Vinci". Cependant, si une autre phrase dans le corpus dit "La Joconde a été peinte par Michel-Ange", le modèle peut aussi bien considérer cette phrase comme valide, car il a appris que les deux artistes étaient associés à ce tableau. Dans ce cas, bien que la réponse de Michel-Ange soit incorrecte d'un point de vue factuel, elle pourrait sembler cohérente

[4] Lukas Berglund et al. - *The Reversal Curse* [6]
[5] Nanjiang Jiang et al. - *Evaluating bert for NLI, case study on the commitmentbank* [7]
[6] Suchin Gururangan et al. - *Annotation Artifacts in Natural Language Inference Data* [9]
[7] Oswald Ducrot - *La théorie des blocs sémantiques* [10]

pour le modèle en fonction de ses apprentissages sur la similarité distributionnelle des mots et des concepts.

3.3 Exploration des approches hybrides

C'est dans un but d'améliorer les performances des modèles de ML qu'interviennent les approches hybrides, qui visent à combiner les avantages du ML avec les capacités expressives de la logique symbolique. La logique symbolique offre un cadre formel pour raisonner sur les relations sémantiques, ce qui permet de capturer les inférences logiques complexes qui échappent souvent aux modèles de ML. En intégrant la logique symbolique, il est possible d'enrichir les modèles de ML avec des règles et des contraintes supplémentaires. Dans l'une ce ces méthodes [8], plusieurs modules sont dédiés à des aspects spécifiques de la compréhension des textes, comme l'identification des relations temporelles et des entités nommées. Ces modules sont ensuite fusionnés en utilisant un système de vote, qui combine les prédictions des modules en tenant compte de leur fiabilité respective, pour produire une décision finale.

D'autres approches hybrides visant à améliorer les résultats obtenus par le ML pour des tâches de RTE/NLI [9] ont été étudiés [10]. Il y a dans un premier temps celles qui viennent renforcer les capacités des modèles de ML en se distinguent par exemple avec l'utilisation de LSTM (Long Short-Term Memory) bidirectionnels [11] et d'un mécanisme d'Attention Spécifique qui permet de gérer efficacement les dépendances entre les mots à longue distance dans le texte [12]. Cette technique se situe à l'intersection du Machine Learning et de la logique symbolique, en ajustant l'importance accordée à chaque mot en fonction de son rôle dans la phrase. Testée sur le corpus de données SNLI [13], cette méthode a surpassé les approches existantes en précision, tout en nécessitant moins de ressources computationnelles.

Une autre approche consiste à explorer les embeddings[14] générés par des modèles de langage pré-entraînés. L'idée est de déterminer si les relations manipulées par ces LLM encodent par défaut les propriétés qui nous intéressent [15]. Par exemple, il est possible d'analyser la représentation de chaque paire d'énoncés sous forme de "tokens CLS", qui sont chargés d'encapsuler l'information pertinente d'une phrase. En procédant ainsi, il serait possible d'effectuer du clustering sur ces représentations et observer si des patterns émergent.

Les embeddings générés par des modèles de langage pré-entraînés permettent d'explorer comment les informations sont traitées dans ces modèles. L'un des plus utilisé est

[8] Mohamed H. Haggag et al. - *Different Models and Approaches of RTE* [11]
[9] RTE/NLI : Recognizing Textual Entailment / Natural Language Inférence
[10] I Made Suwija Putra et al. - *Recognizing textual entailment* [12]
[11] Le modèle analyse la phrase dans les deux sens pour saisir le contexte complet.
[12] Yang Liu et al. - *Learning NLI using Bidirectional LSTM model and Inner-Attention* [13]
[13] Stanford Natural Language Inference [14]
[14] représentations vectorielles de mots ou de phrases dans l'espace
[15] Gabriella Chronis et al. - *A Method for Studying Semantic Construal [...]* [15]

BERT, un modèle de langage pré-entraîné qui fonctionne en utilisant un transformer bidirectionnel pour prendre en compte le contexte précédent et suivant lors de l'encodage des mots dans une phrase. BERT est généralement utilisé en tant que modèle de base pour une variété de tâches de traitement du langage naturel, telles que la classification de texte, la compréhension de texte et la génération de texte. Il est également fine-tuné [16] sur des tâches spécifiques avec des jeux de données annotés afin d'adapter ses capacités à des domaines particuliers et à des objectifs spécifiques de traitement du langage naturel. [17].

Bien qu'il a été montré que des méthodes symboliques de reconnaissance d'inférences textuelles sont un moyen d'améliorer grandement la compréhension du langage naturel par les LLM, cette dernière n'est pas toujours intégrée dans les Grands Modèles de Langue pour améliorer leurs performances.
Plusieurs raisons expliquent cela. D'une part, l'incorporation des inférences textuelles nécessite d'ajouter des algorithmes complexes, ce qui peut entraîner une augmentation significative du temps de calcul [18]. D'autre part, la disponibilité de données annotées pour les inférences textuelles est souvent limitée, ce qui rend difficile l'entraînement de LLM sur ce genre de tâches [19].

[16] Ajustement spécifique d'un modèle pré-entraîné à une tâche particulière après son entraînement initial

[17] Jacob Devlin et al. - *BERT : Pre-training of Deep Bidirectional Transformers for Language Understanding* [16]

[18] Mohamed H. Haggag et al. - *Different Models and Approaches of RTE* [11]

[19] Jiyi Li - *A Comparative Study on Annotation Quality of LLM via Label Aggregation* [17]

4. Axes de travail à venir

4.1 Description de la méthode hybride envisagée

Pour la suite de ce stage, la méthode hybride envisagée vise à développer un système capable de prédire les différentes dimensions annotées des exemples de notre corpus de données, à savoir la similarité argumentative et les relations d'implications. Pour ce faire, nous commencerons par entraîner un modèle de Machine Learning sur les données de RTE/NLI afin d'évaluer ses performances pour ces tâches spécifiques. Nous nous attendons à ce que le modèle présente de meilleures performances pour les relations de similarité et de ciblage argumentatifs que pour les relations d'implications, en raison de la nature complexe et nuancée de ces dernières.

4.2 Collecte et annotation des données pour le corpus

Une première approche pour constituer un corpus serait de repartir des données du CommitmentBank [7], et examiner les différents contextes dans lesquels des phrases sont utilisées, ainsi que les caractéristiques de ces contextes qui semblent influencer la façon dont les phrases sont interprétées ou perçues. Il s'agirait ensuite de regarder les environnements d'un point de vue argumentatif, et de déterminer si ces contextes modifient ou non la direction de l'argumentation exprimée dans la phrase associée. Par exemple, une phrase contenant l'expression "je pense que ..." devrait normalement maintenir la même direction argumentative que son complément, bien que la force de l'argument puisse être légèrement atténuée sans pour autant changer radicalement le sens global de l'argument, alors que pour d'autres prédicats cela sera sans doute diffèrent.
Cette approche, bien que traitant avec un corpus en Anglais, pourrait constituer un bon point de départ pour l'analyse des marqueurs discursifs, mais s'éloigne un peu du sujet dans le sens où nous ne contrôlons pas les prédicats qui ont été extraits (qui sont ici essentiellements des verbes), et qui ne sont pas aussi prototypiques que les opérateurs mentionnés dans cette synthèse, à savoir "presque" et "à peine".

Une autre possibilité est d'aller extraire des phrases contenant ces adverbes en francais (comme par exemple dans Wikipedia, Common Crawl, etc...), et utiliser des techniques de TALN standard pour faire du découpage de phrases, et aller récupérer celles qui contiennent les opérateurs souhaités. Nous devons également s'assurer que les phrases contiennent un seul opérateur, pour éviter de compliquer la tâche en ayant des phrases avec par exemple "presque" ainsi qu'une négation, ce qui augmenterait les interactions.

Une fois les phrases extraites, il s'agirait ensuite d'en extraire le préjacent, là où dans la plupart des cas il peut être juste question d'effacer l'opérateur ; "j'ai presque fini" donne "j'ai fini" pour avoir la paire qui nous intéresse.

À l'aide des paires extraites, il serait possible d'annoter ces dernières de manière semi-automatique. Avec les phrases "j'ai fini ma biere" et "j'ai a peine fini ma bière", nous savons que d'un point de vue sémantique, une des deux implique la négation de l'autre, et nous pouvons donc étiqueter de la relation d'inférence.
Il peut également être envisagé d'entraîner un modèle pour voir s'il est capable de faire de l'annotation tout seul, mais cette piste reste à étudier concernant sa pertinence.

4.3 Mise en place du système de prédiction

Après l'extraction du corpus, plusieurs possibilités pour poursuivre le stage sont envisageables. Dans le cas où les données seraient toutes homogènes, il serait alors pertinent de s'orienter vers du clustering, et de voir comment les représentations s'organisent les unes par rapport aux autres. L'idée serait alors de reprendre le concept évoqué dans l'état de l'art [15], et d'aller regarder la représentation de chaque paire de "phrase-préjacent" dans un modèle de langue pré-entraîné comme BERT, puis de voir si l'on observe des clusters. Le résultat attendu est que toutes les paires soient assez proche entre elles au niveau du token CLS, car n'ayant qu'un seul mot de différence. De plus, il est également attendu que les paires qui sont co-orientées, ou du moins plus proches argumentativement, forment de meilleurs clusters mieux que les autres.
Les premiers tests effectués nous ont permis d'obtenir le résultat préliminaire suivant, avec en rouges la distance cosinus entre chaque paire.

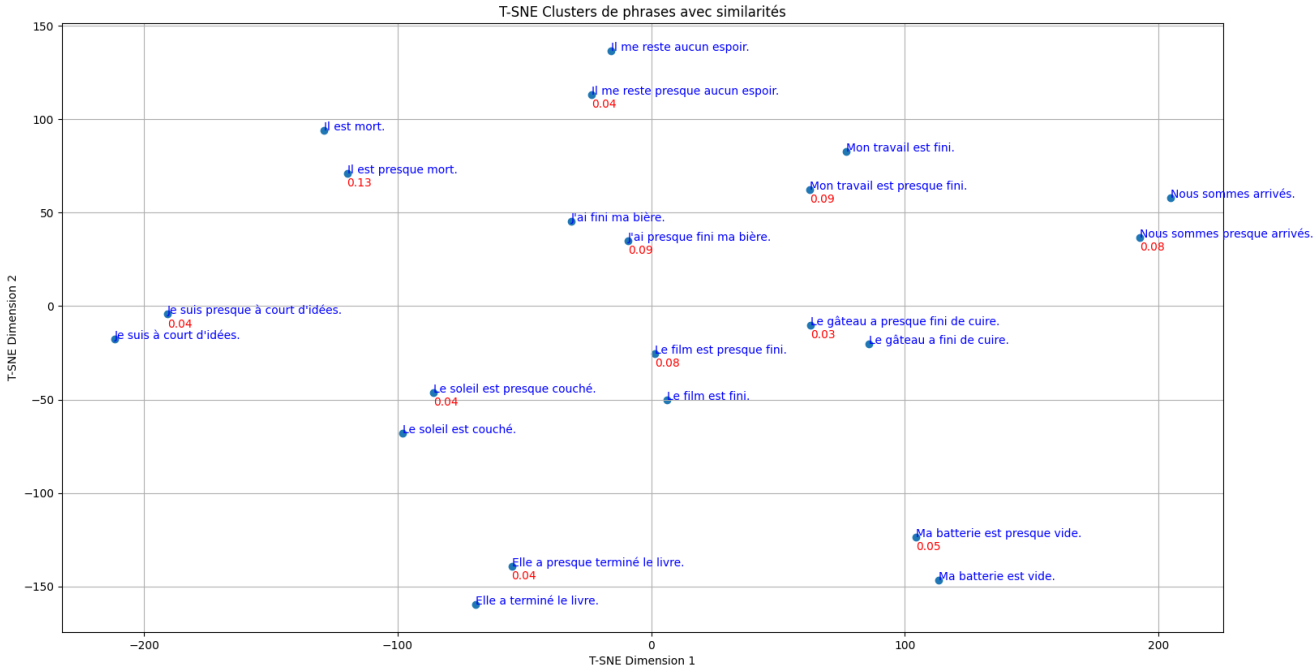

FIGURE 4.1 : Visualisation des distances sémantiques entre des paires de phrases.

En revanche, s'il s'avère que pour un opérateur donné, il semble exister une variété de profils argumentatifs, dépendant crucialement du contexte d'utilisation des phrases, il serait intéressant d'évaluer les performances d'un modèle fine-tuned pour ces spécificités.

Par la suite, nous explorerons comment intégrer une approche symbolique dans le système pour compléter et améliorer les prédictions du modèle de ML. Cette approche symbolique consistera à utiliser des règles et des représentations linguistiques pour guider le processus d'inférence et enrichir la compréhension des relations argumentatives. Nous examinerons également comment ces deux approches peuvent être combinées pour obtenir des résultats plus robustes, notamment à l'aide du système de vote [11].

En perspective, nous avons également discuté la possibilité de générer des données à partir de modèles génératifs comme BERT ou GPT-3.5, et de les comparer à des données réeles. Cette comparaison nous permettrait d'évaluer si les données générées présentent des similitudes avec le corpus initial. Cependant, il est important de souligner que cette démarche constituerait une question de recherche distincte, ne relevant pas directement du sujet principal de l'étude

Bibliographie

[1] Ido Dagan, Dan Roth, Mark Sammons, and Fabio Zanzotto. *Recognizing Textual entailment : models and applications*. Morgan & Claypool Publishers, 2013.

[2] Jean-Claude Anscombre and Oswald Ducrot. *L'argumentation dans la langue*. Pierre Mardaga, Liège, Bruxelles, 1983.

[3] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE : A multi-task benchmark and analysis platform for natural language understanding. *CoRR*, abs/1804.07461, 2018.

[4] Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. Superglue : A stickier benchmark for general-purpose language understanding systems. *CoRR*, abs/1905.00537, 2019.

[5] Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. Siren's song in the AI ocean : A survey on hallucination in large language models. *arXiv preprint arXiv :2309.01219*, 2023.

[6] Lukas Berglund, Meg Tong, Max Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. The Reversal Curse : LLMs trained on "A is B" fail to learn "B is A", 2023.

[7] Nanjiang Jiang and Marie-Catherine de Marneffe. Evaluating bert for natural language inference, a case study on the commitmentbank. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP 2019)*, pages 6086–6091, 2019.

[8] OpenAI. Chatgpt : A large-scale generative model for open-domain chat. `https://github.com/openai/gpt-3`, 2021.

[9] Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. Annotation artifacts in natural language inference data. 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 107–112, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[10] Oswald Ducrot. La théorie des blocs sémantiques. *Verbum*, XXXVII(1–2) :53–65, 2016.

[11] Mohamed Hassan Haggag, Marwa M. A. Elfattah, and Ahmed Mohammed Ahmed. Different models and approaches of textual entailment recognition. *International Journal of Computer Applications*, 142 :32–39, 2016.

[12] I Made Suwija Putra, Daniel Siahaan, and Ahmad Saikhu. Recognizing textual entailment : A review of resources, approaches, applications, and challenges. *ICT Express*, 10(1) :132–155, 2024.

[13] Yang Liu, Chengjie Sun, Lei Lin, and Xiaolong Wang. Learning natural language inference using bidirectional lstm model and inner-attention, 2016.

[14] Anish Mishra. Deep learning techniques in textual entailment. 2018.

[15] Gabriella Chronis, Kyle Mahowald, and Katrin Erk. A method for studying semantic construal in grammatical constructions with interpretable contextual embedding spaces, 2023.

[16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert : Pre-training of deep bidirectional transformers for language understanding, 2019.

[17] Jiyi Li. A comparative study on annotation quality of crowdsourcing and llm via label aggregation, 2024.