Actes de CORIA-TALN 2026
Nantes, France, 29 juin au 3 juillet 2026

Étude des propositionneurs multilingues : formalisation, évaluation et interprétabilité

Luc Pommeret[1] Thomas Gerald[1] Christophe Servan[2] Sahar Ghannay[1]
Patrick Paroubek[1] Sophie Rosset[1]
(1) Université Paris-Saclay, CNRS, LISN, 91400, Orsay, France
(2) AMIAD, Pôle Recherche, 91120, Palaiseau, France
prénom.nom@lisn.fr

RÉSUMÉ

Les propositions atomiques sont de plus en plus utilisées en traitement automatique des langues (TAL) et en recherche d'information (RI), sans qu'il y ait accord sur une définition formelle. Nous proposons un cadre théorique basé sur la théorie de l'information sémantique de Carnap et Bar-Hillel, montrant qu'une proposition atomique correspond exactement à une clause en forme normale conjonctive. Nous entraînons `MPropositionneur-V2`, un propositionneur multilingue distillé de `Qwen3-72B` à `Qwen3-0.6B`, doté d'un algorithme récursif favorisant l'atomicité des sorties. Notre modèle dépasse celui de (Chen *et al.*, 2024), tout en étant plus petit et multilingue. Nous l'évaluons sur trois tâches (extraction de triplets, recherche d'information, évaluation de résumés). Il améliore le rappel des relations, dépasse la granularité des phrases et passages en RI, et est cinq fois plus rapide sur SummEval, avec de meilleurs scores de cohérence et de pertinence.

ABSTRACT

Study of Multilingual Propositioners : Formalisation, Evaluation and Interpretability

Atomic propositions are increasingly used in natural language processing (NLP) and information retrieval (IR), but there is no consensus in the literature on a formal definition. We therefore propose a theoretical framework based on Carnap and Bar-Hillel's semantic information theory, from which we establish that an atomic proposition corresponds exactly to a clause in conjunctive normal form. We then train `MPropositionneur-V2`, a multilingual propositionalizer distilled from `Qwen3-72B` to `Qwen3-0.6B`, using a recursive algorithm that favors atomicity in the outputs. Our model outperforms the propositional generator of (Chen *et al.*, 2024) in terms of ensemble F1 *BERTScore*, while being multilingual and smaller. We evaluate it on three tasks (triplet extraction, information retrieval, and summary evaluation). It improves relation recall in triplet extraction, exceeds sentence and passage granularity in IR, and is five times faster on SummEval, with better consistency and relevance scores.

MOTS-CLÉS : propositions atomiques, RAG, évaluation de résumé, extraction de relations, recherche d'information, propositionneur.

KEYWORDS: Atomic Propositions, RAG, summary evaluation, relation extraction, information retrieval, propositioner.

Actes de la 21e Conférence en Recherche d'Information et Applications.
Volume 1 : articles scientifiques originaux, pages 179–196.
Cette œuvre est mise à disposition sous licence Attribution 4.0 International.

Introduction

Les approches actuelles d'explicabilité restent largement *post hoc*, car celles-ci tentent généralement d'expliquer des décisions déjà prises par des modèles « boîtes noires ». L'approche que nous proposons dans ce travail s'inscrit dans le cadre de l'interprétabilité par construction, où la structure des données est traçable. Pour cela, nous proposons d'étudier l'utilisation d'une structure élémentaire (proposition atomique) pour augmenter l'interprétabilité des systèmes sans en sacrifier la performance. Par exemple, la phrase « Le chat et le chien sont dans la cuisine » peut être décomposée en deux propositions atomiques : « Le chat est dans la cuisine » et « Le chien est dans la cuisine ». La décomposition atomique (ou atomisation) pourrait avoir un impact positif sur un large panel de tâches. De nombreux cas d'utilisation de ces unités linguistiques ont été précédemment étudiés dans la littérature du TAL (Chen *et al.*, 2024; Min *et al.*, 2023; Herserant & Guigue, 2025), montrant un intérêt croissant pour les approches recourant à ce type de décomposition.

En particulier, pour l'extraction d'informations libres (*OpenIE*), l'utilisation de propositions atomiques pourrait améliorer les performances tout en garantissant des décisions interprétables. Dans cet article, nous proposons d'entraîner et d'étudier un modèle de propositionneur multilingue, c'est-à-dire un modèle qui génère une liste de propositions atomiques à partir d'un texte. Cependant, les travaux précédents exploitant les propositions atomiques, en particulier les travaux de Herserant & Guigue (2025), s'appuient sur la construction de propositions au travers du requêtage d'un modèle génératif. De plus, cette précédente approche utilise *via prompting* un grand modèle de langue génératif (`Qwen2.5-72B`) non spécialisé pour l'extraction de propositions. Nous pensons qu'un modèle entraîné sur la tâche de décomposition en atomes pourrait permettre un gain substantiel en termes de précision du modèle. Par ailleurs, l'utilisation de très grands modèles de langue est un facteur limitant pour une adaptation (fine-tuning) et l'inférence pour des ressources de calcul modérées. Ceci étant, ces approches, bien qu'efficaces, ne permettent pas, d'après des expériences réalisées en amont, de garantir l'atomicité des décompositions, en particulier la propriété d'indivision des atomes, c'est-à-dire qu'un atome ne peut être divisé en sous-propositions sans en changer la sémantique.

L'hypothèse que nous souhaitons valider est qu'il serait possible d'améliorer les performances de tâches de TAL et de RI par l'utilisation de propositions simples (respectant la propriété d'indivision). Afin de favoriser la création de propositions atomiques *stricto sensu*, nous apportons plusieurs modifications dans la construction du propositionneur, en particulier *via* l'ajout d'un raffinement successif des propositions générées. Aussi, nous proposons une méthode se basant sur la distillation de grands modèles (Hinton *et al.*, 2015), afin de produire un propositionneur à impact computationnel réduit, cette propriété est particulièrement souhaitable du fait de l'utilisation du raffinement successif (appelant de multiples fois le modèle génératif sur ses propres sorties). Dans cet article, nous explorons l'impact de notre propositionneur en évaluant sa performance sur différentes tâches : l'extraction de triplets entités-relation, la recherche d'information sur deux benchmarks de la littérature et enfin, l'évaluation de résumés. Nous démontrons empiriquement que le propositionneur construit se révèle particulièrement adapté pour les différentes tâches sus-nommées.

Notre article est organisé comme suit : la section 1 dresse un état de l'art des propositions atomiques et de leurs cas d'usages. Nous définissons formellement ce que nous considérons comme proposition atomique dans la section 2, en s'appuyant sur la théorie de l'information sémantique. La section 3 définit l'entraînement et le fonctionnement de notre propositionneur[1]. Enfin, les sections 4 et 5 détaillent respectivement le protocole expérimental et les résultats obtenus sur différentes applications.

1. Le propositionneur est disponible ici : `https://huggingface.co/Zual/MPropositionneur-V2`

1 État de l'art

Atomes et applications. Les tâches d'extraction d'information (EI), de Recherche d'Information (RI) et de vérification de faits (fact-checking) sont traditionnellement abordées en fournissant directement des documents longs ou des phrases complexes. Par exemple en RI, les paragraphes sont souvent fournis directement à un modèle de plongement pour en obtenir leurs représentations vectorielles. En RI, l'indexation de propositions atomiques plutôt que de passages entiers permet une récupération plus fine tout en explicitant et en localisant l'information pertinente, mais améliore également, dans certains cas, les performances d'un système (Chen *et al.*, 2024). Pour l'inférence en langue naturelle (NLI), décomposer les prémisses en atomes (textes sur lequel s'appuyer pour décider de la validité d'une hypothèse) s'est montré efficace (Stacey *et al.*, 2024). Aussi, il est plus aisé de vérifier des passages peu denses en informations plutôt que de paragraphes composés d'informations multiples (Min *et al.*, 2023). Cette décomposition est donc particulièrement adaptée à la tâche de vérification des faits (Fact-Checking). De même, en évaluation de résumé, les méthodes basées sur ces décompositions atomiques ont démontré des corrélations avec l'évaluation humaine (Herserant & Guigue, 2025). Le présent article s'inscrit dans cette continuité en explorant l'utilisation polyvalente d'un « *propositionneur* » générant des propositions atomiques pour l'explicabilité, dans le cadre de plusieurs tâches liées à la RI et l'EI.

Propositionneur. À notre connaissance, le terme « *propositionizer* » est mentionné pour la première fois dans Chen *et al.* (2024) qui ont développé un générateur de propositions atomiques en affinant `Flan-T5-large`. Celui-ci a été réutilisé pour servir de base de comparaison par Wang *et al.* (2025) afin d'évaluer des méthodes de segmentation de documents pour la RI. You *et al.* (2025) utilisent le même principe en combinant cette fois plusieurs niveaux de segmentation sémantique allant de l'entité au segment de texte en passant par la phrase pour produire des représentations.

Pour accélérer la recherche d'information, Guo *et al.* (2025) exploitent les propositions atomiques plongées dans un espace discret conjointement aux requêtes, en faisant l'hypothèse qu'il est plus facile dans un espace discret fini de représenter des propositions atomiques que de longs textes.

Kalra *et al.* (2025) ont utilisé le propositionneur pour sélectionner dynamiquement, en fonction d'une requête, des combinaisons pondérées d'extracteurs hétérogènes de documents. Les requêtes et les documents sont décomposés en unités sémantiques atomiques alignées à l'aide d'une approche « à un exemple » (*one-shot*) pour leurs expériences de RI.

Les travaux de Li *et al.* (2025) utilisent les propositions atomiques afin de nettoyer des données synthétiques d'apprentissage pour le raisonnement en plusieurs étapes. L'avantage de cette méthode repose sur l'encapsulation des unités d'information factuelles (*factoid*) en langue naturelle pour la RI.

Toutes ces approches prouvent donc empiriquement l'intérêt des représentations en atomes, néanmoins, la notion même de proposition atomique reste à définir, celle-ci variant selon les travaux réalisés.

Propositions atomiques. Le concept de proposition atomique est historiquement inscrit dans le cadre de l'atomisme logique théorisé par Russell & Pears (1918). Russell définit la proposition atomique comme l'expression d'un *fait atomique*, lequel est constitué d'une relation (monadique, dyadique ou n-adique) reliant des termes appelés *particuliers*. Cela revient en fait à considérer les propositions dépourvues de connecteurs logiques, cependant cette définition a évolué depuis.

La question de l'atomicité des représentations sémantiques n'est pas nouvelle en traitement auto-

matique des langues, par exemple Keenan & Faltz (1985) mentionnent la propriété d'atomicité des algèbres booléennes pour représenter la sémantique de la langue naturelle. Johnson & Kay (1990) proposent la notion d'« infon » correspondant à une proposition atomique pour une situation donnée dans une représentation sémantique guidée par la syntaxe. D'autre part, l'atomicité du concept dénoté est une question essentielle dans la définition de la télicité des verbes (Krifka, 1998; Caudal, 1999)[2].

Récemment, la notion de proposition atomique a été reprise dans divers travaux en TAL avec un glissement de sens qu'il convient de préciser. Notamment pour la vérification de fait, Lu *et al.* (2025) proposent une granularité adaptative de décomposition des faits en fonction de l'atomicité préférentielle du vérificateur de faits, plutôt que de considérer une granularité sémantique statique.

Chen *et al.* (2024) définissent les propositions comme des « expressions atomiques au sein d'un texte, où chacune exprime un fait distinct (*factoid*) et se présente sous un format de langue naturelle concis et auto-suffisant (*self-contained*) ». Pour (Min *et al.*, 2023), les propositions atomiques sont des « énoncés courts qui contiennent chacun une seule unité d'information (*piece of information*) ».

Hosseini *et al.* (2024) donnent une liste de critères que doivent satisfaire les propositions : « être atomique et constituer une unité sémantique minimale [...] ne peut pas être plus décomposée en d'autres unités ayant un sens ». Enfin, Wanner *et al.* (2024) font une étude systématique des différentes approches des propositions atomiques.

Nous pouvons donc voir que la littérature ne s'accorde pas sur une définition unique du concept de proposition atomique, ainsi, dans la section suivante, nous décrivons formellement le concept que nous convoquons ici.

2 Propositions atomiques

En accord avec les travaux mentionnés précédemment, nous définissons une proposition atomique comme une proposition qui respecte les deux propriétés fondamentales suivantes :

1. ATOMICITÉ. Elle est la plus petite unité sémantique telle que toute nouvelle décomposition de celle-ci constituerait une augmentation dans l'**information sémantique** (intuitivement, décomposer une proposition atomique impliquerait un changement sémantique).
2. AUTONOMIE. Elle est sémantiquement complète. Toute entité (ou *particulier*) doit être explicitée par son nom propre ou une description univoque.

Dans le cadre de nos travaux, nous proposons une formalisation de cette définition en nous appuyant pour cela sur le formalisme de la théorie de l'information sémantique (Bar-Hillel & Carnap, 1953). Le cœur de cette théorie est de transférer la théorie de l'information de Shannon dans le domaine de la sémantique, c'est-à-dire, celui de la logique des prédicats du premier ordre. Soit ϕ une formule de la logique des prédicats du premier ordre. Nous pouvons dresser la table de vérité de ϕ, et considérer les mondes qui satisfont la formule. Nous appelons $\mathrm{Cont}(\phi)$ cet ensemble :

$$\mathrm{Cont}(\phi) = \{w \in \mathcal{W} | w \models \phi\}$$

Où l'ensemble $\mathcal{W}$ est l'ensemble des mondes possibles, c'est-à-dire l'ensemble des lignes de la table de vérité de ϕ. L'information de la formule ϕ est alors définie comme l'information au sens

2. La télicité est une dimension *aspectuelle* d'un verbe ou d'une expression verbale qui indique que sa sémantique dénote un événement ou une action qui comprend un aboutissement, une fin définie.

de Shannon, c'est-à-dire la proportion de mondes qui satisfont ϕ : $I(\phi) = -\log_2\left(\frac{|\text{Cont}(\phi)|}{|\mathcal{W}|}\right)$ avec $\frac{|\text{Cont}(\phi)|}{|\mathcal{W}|}$ la proportion des mondes satisfaisant ϕ.

Exemple (A∧B). *Ici,* $\text{Cont}(A \wedge B) = \text{Cont}(\phi) = \{w \in \mathcal{W} | w \models A \wedge B\} = \{(1,1)\}$ *et* $\mathcal{W} = \{(0,0);(0,1);(1,0);(1,1)\}$. *L'information sémantique contenue dans* $A \wedge B$ *est de 2 :* $I(A \wedge B) = -\log_2\left(\frac{1}{4}\right) = 2$

Nous introduisons ensuite un critère de décomposition d'une proposition. La décomposition sera dite « sûre » si elle diminue l'information :

Définition 4 (Décomposition sûre). *La décomposition d'une formule* ϕ *en une formule* ψ *est dite* ***sûre*** *(resp.* ***strictement sûre****) si* ψ *est une sous-formule de* ϕ *et que* $I(\phi) \geq I(\psi)$ *(resp.* $I(\phi) > I(\psi)$*).*

Au contraire, une décomposition sera dite « mauvaise » si elle augmente ou conserve l'information :

Définition 5 (Mauvaise décomposition). *La décomposition d'une formule* ϕ *en une formule* ψ *est dite* ***mauvaise*** *si* ψ *est une sous-formule stricte de* ϕ *et que* $I(\phi) \leq I(\psi)$

Ces définitions permettent d'établir les propriétés suivantes, dont les preuves sont développées en Annexe C :

Lemme 1 (Conjonction). *La décomposition d'une formule* $A \wedge B$ *en une formule* A *est sûre.*

Lemme 2 (Disjonction). *La décomposition d'une formule* $A \vee B$ *en une formule* A *est mauvaise.*

Lemme 3 (Implication). *La décomposition d'une formule* $A \rightarrow B$ *en une formule* A *est mauvaise.*

Ces résultats s'interprètent naturellement. Dire « Le chat et le chien sont dans la cuisine » contient plus d'information que « Le chat est dans la cuisine » : la décomposition conjonctive est sûre. En revanche, dire « Le chat ou le chien est dans la cuisine » est moins informatif que « Le chat est dans la cuisine » : la décomposition d'une disjonction hallucine une information plus précise que ce qui était affirmé. De même, si l'on dit « S'il pleut, le sol est mouillé », affirmer qu'il pleut ajoute de l'information par rapport à l'implication d'origine.

La proposition atomique est donc la plus petite unité formée par décomposition à partir d'un texte telle que toute nouvelle décomposition de celle-ci serait mauvaise. Nous avons établi que cette caractérisation coïncide exactement avec les clauses en Formes Normales Conjonctives (FNC) :

Définition 3 (Proposition atomique). *Une proposition est dite* ***atomique*** *si elle est une formule close (*AUTONOMIE*) et qu'on ne peut pas la décomposer plus sans faire une mauvaise décomposition (*ATOMICITÉ*).*

Théorème 1 (Caractérisation structurelle des propositions atomiques). *Une proposition est atomique si et seulement si elle est une clause d'une forme normale conjonctive (FNC)*[3].

Une conséquence pratique est qu'une proposition atomique ne peut pas être décomposée en conjonction de sous-faits plus simples. En revanche, elle peut contenir une disjonction (« A ou B »), car décomposer une disjonction reviendrait à halluciner une information plus précise que celle exprimée dans le texte original comme dans l'exemple cité précédemment.

3. La preuve de ce théorème est en Annexe C.

3 Entraînement d'un propositionneur

Pour transférer notre définition dans le cadre de la langue naturelle, il est nécessaire que les propositions générées à partir d'un texte soient atomiques. Pour ce faire, nous avons un schéma en deux étapes. Dans un premier temps, nous nous appuyons sur les travaux de (Chen *et al.*, 2024) via l'entraînement d'un propositionneur se basant sur la distillation d'un grand modèle de langue. Dans un second temps, nous ajoutons des garde-fous pour favoriser une bonne décomposition en propositions atomiques (1) par vérification de la génération lors de la distillation (voir prompt A.2) (2) par utilisation récursive du propositionneur (voir l'algorithme 1) et (3) par évaluation des propositions atomiques avec différentes métriques.

Génération d'exemples d'entraînement. Pour produire les propositions atomiques qui serviront au processus de distillation, nous procédons en deux étapes :

1. nous faisons une première passe de génération avec `Qwen3-72B` (voir prompt en Annexe A.1)
2. puis une seconde passe en donnant la proposition atomique ainsi que le contexte au même `Qwen3-72B`

La seconde étape permet de vérifier si la proposition est bien autonome (voir le prompt en Annexe A.2).

Dans leur expérience, Chen *et al.* (2024) affinent `Flan-T5-large` et produisent `propositionizer-wiki-flan-t5-large`[4], qui est suffisant pour leurs expériences en **RAG**. Cependant, la taille de contexte de 512 tokens apparaît vite comme insuffisante lorsque l'on a des documents de plus grande taille, avec des dépendances contextuelles longues, comme dans le corpus CQuAE (Gerald *et al.*, 2024).Nous avons donc choisi de passer à une architecture uniquement décodeur (*decoder-only*), dotée d'un plus long contexte de 32 768 tokens, avec `Qwen3-72B`. En sortie du modèle, nous récupérons les propositions atomiques au format `json`. La sortie est filtrée pour ne retenir que les propositions correctement formatées en `json`.

Ressources multilingues. Nous produisons des propositions atomiques pour l'entraînement en utilisant des passages de Wikipédia[5]. Concernant les langues, nous nous sommes limités à six langues européennes occidentales, à savoir l'anglais, le français, l'allemand, l'italien, l'espagnol et le portugais. Nous avons récupéré 56 000 passages de Wikipédia, suivant la méthodologie de Chen *et al.* (2024), en collectant des passages limités à 2000 tokens. Cette limite arbitraire, très inférieure à la taille maximale de contexte de Qwen3-0.6B (32 768 tokens), a été choisie pour tenir compte du facteur d'expansion du nombre de tokens lié à la décomposition des propositions atomiques.

Nous définissons S l'ensemble des passages récupérés depuis Wikipédia. Nous appelons P_{train} le sous-ensemble de propositions atomiques obtenues par le modèle enseignant sur S. Nous utilisons ce sous-ensemble pour adapter le modèle apprenant `Qwen3-0.6B` selon la méthode décrite par (Hinton *et al.*, 2015).

Métriques d'évaluation des propositions générées. Comme dans Chen *et al.* (2024), nous utilisons pour l'évaluation un score BERT (*BERTScore*) ensembliste :

$$\text{Pr} = \frac{1}{|\hat{P}_i|} \sum_{p \in \hat{P}_i} \max_{p' \in P_i} (cos(p, p')) \quad \text{Ra} = \frac{1}{|P_i|} \sum_{p' \in P_i} \max_{p \in \hat{P}_i} (cos(p, p')) \quad \text{F1} = \frac{2 \times \text{Pr} \times \text{Ra}}{\text{Pr} + \text{Ra}} \tag{1}$$

4. https://huggingface.co/chentong00/propositionizer-wiki-flan-t5-large
5. https://huggingface.co/datasets/wikimedia/wikipedia

Avec P_i l'ensemble des propositions atomiques obtenue par le modèle enseignant et $\hat{P}_i$ l'ensemble des propositions atomiques obtenues par le modèle apprenant sur le i-ème passage où N est le nombre de passages.

Par défaut, le score BERT ensembliste est borné par la validité du format `json` de sortie, car on ne peut pas identifier les propositions d'une liste `json` qui serait syntaxiquement incorrecte. Pour pallier ce problème, nous limitons donc le score BERT ensembliste aux structures `json` qui sont syntaxiquement correctes.

Pour calculer l'écart de densité informationnelle entre ce qui est attendu et ce qui est prédit, nous utilisons également d'autres métriques, pour le décompte des propositions, en particulier la MAE (*Mean Absolute Error*), ainsi qu'un biais simple :

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N} ||\hat{P}_i| - |P_i|| \quad \text{biais} = \frac{1}{N}\sum_{i=1}^{N} |\hat{P}_i| - |P_i| \tag{2}$$

Propositionneur récursif. Nous postulons qu'une meilleure extraction de propositions atomiques au sens défini dans la section précédente (atomicité) pourrait impacter positivement les performances sur les tâches cibles. Nous proposons à cette fin de raffiner les sorties de notre propositionneur. Le principe consiste à réitérer la décomposition sur les propositions de sorties précédentes, jusqu'à la convergence, c'est-à-dire qu'aucune nouvelle proposition n'est générée, ou bien que le nombre d'itérations maximum est atteint (voir l'algorithme 1). Notez que la pertinence du seuil de récursion, qui a été fixé empiriquement à 5 suite à nos premières expérimentations, reste à valider par rapport à d'autres modes d'optimisation, comme par exemple l'emploi d'une mesure de similarité sémantique calculée dans un espace de plongement comme dans (Malandri *et al.*, 2025).

Algorithm 1 Propositionneur

Require: $N = 5$, t est un texte, $i \in \mathbb{N}$, $\mathcal{M}$ est le propositionneur

$\mathcal{P}_i$, $\mathcal{A}_i$, l'ensemble des propositions en langage naturel.
$i \leftarrow 0$; $\mathcal{P}_0 \leftarrow \mathcal{M}(t)$; $\mathcal{A}_0 \leftarrow \{p \in \mathcal{P}_0,\ \ \{p\} = \mathcal{M}(p)\}$
while $(\mathcal{P}_i \neq \mathcal{A}_i) \wedge (i < N)$ **do**
    $\mathcal{P}_{i+1} \leftarrow \bigcup_{x \in \mathcal{P}_i} \mathcal{M}(x)$
    $\mathcal{A}_{i+1} \leftarrow \{x \in \mathcal{P}_{i+1},\ \ \{x\} = \mathcal{M}(x)\}$
    $i \leftarrow i + 1$
end while
return $\mathcal{A}_i$

4 Protocole expérimental

Nous évaluons le propositionneur sur trois tâches distinctes, l'extraction de triplets, la recherche d'information et l'évaluation automatique de résumé.

Extraction de triplets en vocabulaire fermé. Pour cette tâche, nous évaluons notre propositionneur sur les deux jeux de données **SMiLER** et **FewRel**, étant donné un passage et un couple d'entités, la tâche consiste à identifier la relation liant ces entités. L'approche *Directe* localise les entités dans le paragraphe source ; l'approche *Proposition* les localise dans les atomes extraits ; l'approche *Combinée* bascule sur les atomes lorsque les entités ne sont pas localisées dans le paragraphe. Enfin, nous évaluons les métriques d'exactitude (*Acc*), étant le nombre de triplets correctement extraits ; le

pourcentage d'entités extraites (*e-rec*) et le pourcentage de relations extraites. Étant en vocabulaire fermé, nous donnons à `GLiREL` (et aux modèles `Qwen3`) l'ensemble des relations du vocabulaire. Pour les entités, nous considérons la correspondance parfaite.

Recherche d'information. Pour la recherche d'information, nous comparons deux corpus en utilisant le modèle de RI *Contriever*[6]. Le premier corpus considéré est *Natural Questions* (Kwiatkowski *et al.*, 2019), le second est le corpus CQuAE v2, un corpus de question-réponse sur des livres scolaires (Gerald *et al.*, 2024). Notons que dans ce dernier corpus il existe une taxonomie des questions : Factuel, la réponse est une entité nommée ; Définition : la réponse est une définition d'un terme du cours ; Cours : la réponse peut être directement extraite d'un passage du cours ; Synthèse : La réponse implique la comparaison d'éléments du cours. Pour les différentes expériences, nous indexons soit les passages, les phrases ou les atomes extraits du corpus. Enfin, nous mesurons le rappel@5, 10 et 20, ainsi que le NDCG@10.

Évaluation de résumé. Pour cette dernière tâche, nous comparons notre méthode à celle de (Herserant & Guigue, 2025). Elle consiste en deux étapes, (1) atomiser le résumé et le texte original, c'est à cet endroit que nous remplaçons le LLM utilisé par (Herserant & Guigue, 2025) par notre `MPropositionneur-V2` (2) évaluer l'adéquation de chacun des atomes du résumé avec les atomes du texte original, permettant ensuite d'étudier la corrélation avec le jugement humain sur un jeu de données annoté par des experts (Fabbri *et al.*, 2021).

Études comparatives. Pour chaque tâche, nous prenons un système à l'état de l'art auquel nous ajoutons en amont le propositionneur. Pour la tâche de recherche d'informations, nous utilisons `contriever-msmarco` comme base de comparaison. Pour la tâche d'évaluation de résumé, nous utilisons le modèle Qwen3-32B.

Métriques. Pour la tâche d'extraction de triplets, nous utilisons les métriques standard suivantes, précision, rappel, score-F1 et AUC. Pour l'évaluation de la recherche d'information, nous utilisons le rappel à k et le NDCG qui sont également standards. Pour l'évaluation de résumé, nous utilisons la métrique de corrélation ρ de Spearman avec les catégories du dataset SummEval (Fabbri *et al.*, 2021; Herserant & Guigue, 2025).

5 Résultats et analyse

Choix du modèle du propositionneur. Nous avons comparé `Flan-T5-large`, `Qwen2.5-0.5B` et `Qwen3-0.6B`. C'est le propositionneur basé sur `Qwen3-0.6B` qui donne le meilleur résultat, au bout de 2 époques, en termes de F1 du score BERT ensembliste (voir section 1). À partir de maintenant, toutes les expériences sont faites avec le `MPropositionneur-V2` issu de l'affinage de `Qwen3-0.6B`.

modèle	JSON valide (%)	MAE	biais	précision	rappel	F1	paramètres
`MPropositionneur-V2` (Qwen3-0.6B)	**93.0**	6.06	**-4.44**	**0.841**	**0.843**	**0.842**	596 042 752
`propositionizer` (Flan-T5-large)	90.0	**5.61**	-5.41	0.829	0.828	0.829	**783 150 080**

TABLE 1 – Comparaison des modèles `MPropositionneur-V2` et `propositionizer-wiki-flan-t5-large` (Chen *et al.*, 2024) sur 100 exemples du jeu de validation (`atomized_val.jsonl`). Score BERT ensembliste calculé avec `xlm-roberta-base` selon la méthodologie de (Chen *et al.*, 2024), voir équation 1 de la section 3.

6. https://huggingface.co/facebook/contriever-msmarco

Nous observons dans la table 1 que notre modèle (environ 600 millions de paramètres) a de meilleures performances que le `propositionizer` de (Chen *et al.*, 2024) (environ 800 millions de paramètres). La différence fondamentale entre les deux est que `MPropositionneur-V2` est multilingue contrairement à `propositionizer`. Nous avons évalué notre propositionneur sur la tâche d'extraction de triplets.

Extraction de triplets en vocabulaire fermé. Dans la table 2, nous reportons les résultats obtenus sur les corpus SMiLER et FewRel. On remarquera que les approches via les propositions atomiques permettent d'obtenir des résultats supérieurs pour le rappel calculé sur les relations. *A contrario*, la méthode directe semble plus efficace pour extraire les entités, et ce, pour les deux corpus évalués. Avec l'approche *Combinée*, nous pouvons apercevoir quel serait l'apport des propositions atomiques par rapport à la méthode *Directe*.

		GLiREL			Qwen3-0.6B			Qwen3-4B		
Jeu de données	Méthode	Acc	e-rec	r-rec	Acc	e-rec	r-rec	Acc	e-rec	r-rec
SMiLER	Directe	57.3	**99.5**	57.6	30.8	**100.0**	30.8	**75.7**	**100.0**	75.7
	Proposition	48.5	82.9	58.5	**38.9**	82.9	**46.9**	65.5	82.9	**79.1**
	Combinée	**59.1**†	**99.5**	**59.3**	–	–	–	–	–	–
FewRel	Directe	48.7	**100.0**	48.7	**42.2**	**100.0**	42.2	**67.7**	**100.0**	67.7
	Proposition	40.2	75.8	53.1	40.3	75.8	**53.2**	51.5	75.8	**68.0**
	Combinée	**50.0**†	**100.0**	**50.0**	–	–	–	–	–	–

TABLE 2 – Résultats sur les jeux de données SMiLER et FewRel. † indique que l'amélioration de Comb sur Direct est significative (bootstrap test, $p < 0.05$). L'atomisation augmente significativement le rappel de relations. La précision augmente avec les petits LLMs (`Qwen3-0.6B`)

RAG. Dans le cas du RAG, les propositions atomiques servent à retrouver les passages les plus pertinents pour répondre à une question donnée. L'utilisation de propositions atomiques est proposée par (Chen *et al.*, 2024), qui atomisent au préalable tout Wikipédia avant de retrouver les passages pertinents. Ils montrent que la granularité optimale (entre le passage, la phrase et la proposition atomique) est la proposition atomique. Nous reproduisons ici leurs expériences avec notre propositionneur, et ajoutons le benchmark CQuAE (Gerald *et al.*, 2024), qui est en français, permettant ainsi de vérifier le multilinguisme de notre propositionneur.

Granularité	**R@1**	**R@5**	**R@20**	**NDCG@10**
Passages	7.60	25.14	48.90	0.241
Phrases	12.89	34.49	58.87	0.387
Propositions	**15.16**	**38.67**	**61.53**	**0.428**

TABLE 3 – Performance en récupération sur NQ (sous-ensemble de 2.6 millions) avec Contriever.

	Passage		**Phrase**		**Proposition**	
Catégorie	R@1	R@5	R@1	R@5	R@1	R@5
Global ($n = 512$)	41.0	63.3	41.2	68.8	**50.4**	**70.9**
Factuel ($n = 125$)	**52.8**	67.2	46.4	71.2	**52.8**	**74.4**
Définition ($n = 126$)	36.5	59.5	28.6	55.6	**50.8**	**73.0**
Cours ($n = 123$)	42.3	67.5	48.0	**80.5**	**55.3**	70.7
Synthèse ($n = 125$)	36.8	62.4	**43.2**	**68.0**	44.0	65.6

TABLE 4 – Rappel@1 et Rappel@5 (%) selon la granularité d'indexation (en utilisant contriever-msmarco) et le type de question, sur le jeu de test CQuAE v2. Les propositions sont générées par le modèle `MPropositionneur-V2`.

Dans la table 3, nous comparons la capacité du modèle Contriever pour différentes granularités de texte

et avec les propositions atomiques extraites par notre propositionneur. Pour les différentes métriques évaluées, la granularité au niveau atomique permet d'obtenir les meilleures performances, validant ainsi l'intérêt de l'utilisation d'atomes pour la RI. Dans la table 4, nous reportons les résultats sur le corpus CQuAE, pour des configurations similaires avec les mêmes granularités. Sur les questions simples où la réponse attendue est un fait ou une définition, nous obtenons les résultats les plus élevés avec notre propositionneur. Cependant, pour des questions plus complexes, en particulier les questions amenant une synthèse de différentes informations (ligne Synthèse), les résultats sont plus mitigés, avec cette fois de meilleures performances pour la granularité **Phrase**.

Évaluation de résumé L'an dernier, Herserant & Guigue (2025) ont proposé une méthode d'évaluation de résumés baptisée Seval-ex, qui utilisait les propositions atomiques en ayant pour but de faire une évaluation explicable. Nous reproduisons ici leurs expériences, en remarquant que notre propositionneur atteint les mêmes performances. Nous remarquons que l'utilisation du propositionneur induit une baisse importante en fluidité et exactitude factuelle, mais de meilleurs scores en cohérence et en pertinence. La vitesse est fortement impactée lors du passage au propositionneur : le traitement est plus rapide.

Méthode	Fluidité	Cohérence	Exactitude factuelle	Pertinence	Vitesse (r/s)
`Qwen3-32B`	**0.366**	0.232	**0.527**	0.266	3.7
`MPropositionneur-V2`	0.249	**0.270**	0.437	**0.319**	**20.5**

TABLE 5 – Comparaison de l'extraction de propositions sur le benchmark SummEval (Fabbri *et al.*, 2021). Spearman ρ entre le score de précision et le jugement humain. La vitesse mesure le nombre de résumés par seconde sur un GPU NVIDIA A6000.

6 Limites

Bien que la correspondance exacte entre clauses FNC et sorties en langue naturelle ne soit pas formellement garantie, notre algorithme récursif vise à approximer cette propriété en décomposant les sorties jusqu'au point fixe. Dans de futurs travaux, nous voulons tester l'efficacité de la récursivité et évaluer les propositions atomiques avec une annotation humaine.

7 Conclusion

Nous avons présenté un modèle de propositionneur multilingue qui atomise un texte en propositions simples. La théorie de l'information sémantique a permis de justifier formellement la nécessité d'une structure indivisible. Les évaluations menées montrent qu'exploiter ce format améliore l'extraction d'information via un propositionneur récursif, la recherche de passages (RI) et la robustesse de l'évaluation en résumé. Enfin, la polyvalence de cette méthode ouvre la voie à des systèmes interprétables, *par construction*, importants pour un traitement de l'information transparent dans de multiples langues. L'intérêt suscité par les premiers propositionneurs dans ces travaux récents ouvre des perspectives intéressantes de progrès, basées sur un va-et-vient entre représentations neuronales et représentations symboliques pour la plupart des systèmes de traitement automatique de la langue naturelle. Dans cet objectif, nous souhaitons explorer plus en détail les moyens les plus efficaces d'optimiser la génération de propositions atomiques à l'aide d'approches neuro-symboliques.

Références

BAR-HILLEL Y. & CARNAP R. (1953). Semantic information. *The British Journal for the Philosophy of Science*, **4**(14), 147–157.

CAUDAL P. (1999). Computational lexical semantics, incrementality, and the so-called punctuality of events. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, p. 497–504, College Park, Maryland, USA : Association for Computational Linguistics. DOI : 10.3115/1034678.1034753.

CHEN T., WANG H., CHEN S., YU W., MA K., ZHAO X., ZHANG H. & YU D. (2024). Dense X retrieval : What retrieval granularity should we use ? In Y. AL-ONAIZAN, M. BANSAL & Y.-N. CHEN, Éds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, p. 15159–15177, Miami, Florida, USA : Association for Computational Linguistics. DOI : 10.18653/v1/2024.emnlp-main.845.

FABBRI A. R., KRYŚCIŃSKI W., MCCANN B., XIONG C., SOCHER R. & RADEV D. (2021). SummEval : Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, **9**, 391–409. DOI : 10.1162/tacl_a_00373.

GERALD T., TAMAMES L., ETTAYEB S., PAROUBEK P. & VILNAT A. (2024). CQuAE : Un nouveau corpus de question-réponse pour l'enseignement. In M. BALAGUER, N. BENDAHMAN, L.-M. HO-DAC, J. MAUCLAIR, J. G MORENO & J. PINQUIER, Éds., *Actes de la 31ème Conférence sur le Traitement Automatique des Langues Naturelles, volume 1 : articles longs et prises de position*, p. 50–63, Toulouse, France : ATALA and AFPC.

GUO J., CHEN X., XIA Q., WANG Z., OU J., QIN L., YAO S. & TIAN W. (2025). HASH-RAG : Bridging deep hashing with retriever for efficient, fine retrieval and augmented generation. In W. CHE, J. NABENDE, E. SHUTOVA & M. T. PILEHVAR, Éds., *Findings of the Association for Computational Linguistics : ACL 2025*, p. 26847–26858, Vienna, Austria : Association for Computational Linguistics. DOI : 10.18653/v1/2025.findings-acl.1376.

HERSERANT T. & GUIGUE V. (2025). Seval-ex : Un paradigme basé sur les phrases atomiques pour une évaluation explicable de la qualité des résumés. In F. BECHET, A.-G. CHIFU, K. PINEL-SAUVAGNAT, B. FAVRE, E. MAES & D. NURBAKOVA, Éds., *Actes de la 20e Conférence en Recherche d'Information et Applications (CORIA)*, p. 217–229, Marseille, France : ATALA \\& ARIA.

HINTON G. E., VINYALS O. & DEAN J. (2015). Distilling the knowledge in a neural network. *ArXiv*, **abs/1503.02531**.

HOSSEINI M. J., GAO Y., BAUMGÄRTNER T., FABRIKANT A. & AMPLAYO R. K. (2024). Scalable and domain-general abstractive proposition segmentation. In Y. AL-ONAIZAN, M. BANSAL & Y.-N. CHEN, Éds., *Findings of the Association for Computational Linguistics : EMNLP 2024*, p. 8856–8872, Miami, Florida, USA : Association for Computational Linguistics. DOI : 10.18653/v1/2024.findings-emnlp.517.

JOHNSON M. & KAY M. (1990). Semantic abstraction and anaphora. In *COLING 1990 Volume 1 : Papers presented to the 13th International Conference on Computational Linguistics*.

KALRA J. S., ZHAO X., KIM T. E., CAI F., DIAZ F. & WU T. (2025). MoR : Better handling diverse queries with a mixture of sparse, dense, and human retrievers. In C. CHRISTODOULOPOULOS, T. CHAKRABORTY, C. ROSE & V. PENG, Éds., *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, p. 11971–11990, Suzhou, China : Association for Computational Linguistics. DOI : 10.18653/v1/2025.emnlp-main.601.

KEENAN M. & FALTZ M. (1985). *Boolean Semanatics for Natural Language*, volume 23 de *Synthese Language Library*. Dordrecht/Boston/Lancaster : D. Reidel Publishing Company - Kluwer Academic.

KRIFKA M. (1998). The origins of telicity.

KWIATKOWSKI T., PALOMAKI J., REDFIELD O., COLLINS M., PARIKH A., ALBERTI C., EPSTEIN D., POLOSUKHIN I., KELCEY M., DEVLIN J., LEE K., TOUTANOVA K. N., JONES L., CHANG M.-W., DAI A., USZKOREIT J., LE Q. & PETROV S. (2019). Natural questions : a benchmark for question answering research. *Transactions of the Association of Computational Linguistics*.

LI Y., GU J.-C., WU D., CHANG K.-W. & PENG N. (2025). BRIEF : Bridging retrieval and inference for multi-hop reasoning via compression. In L. CHIRUZZO, A. RITTER & L. WANG, Éds., *Findings of the Association for Computational Linguistics : NAACL 2025*, p. 5464–5485, Albuquerque, New Mexico : Association for Computational Linguistics. DOI : 10.18653/v1/2025.findings-naacl.301.

LU Y., ZIEMS N., DANG H. & JIANG M. (2025). Optimizing decomposition for optimal claim verification. In W. CHE, J. NABENDE, E. SHUTOVA & M. T. PILEHVAR, Éds., *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 5095–5114, Vienna, Austria : Association for Computational Linguistics. DOI : 10.18653/v1/2025.acl-long.254.

MALANDRI L., MERCORIO F., MEZZANZANICA M. & PALLUCCHINI F. (2025). RE-FIN : Retrieval-based enrichment for financial data. In O. RAMBOW, L. WANNER, M. APIDIANAKI, H. AL-KHALIFA, B. D. EUGENIO, S. SCHOCKAERT, K. DARWISH & A. AGARWAL, Éds., *Proceedings of the 31st International Conference on Computational Linguistics : Industry Track*, p. 751–759, Abu Dhabi, UAE : Association for Computational Linguistics.

MIN S., KRISHNA K., LYU X., LEWIS M., YIH W.-T., KOH P., IYYER M., ZETTLEMOYER L. & HAJISHIRZI H. (2023). FActScore : Fine-grained atomic evaluation of factual precision in long form text generation. In H. BOUAMOR, J. PINO & K. BALI, Éds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, p. 12076–12100, Singapore : Association for Computational Linguistics. DOI : 10.18653/v1/2023.emnlp-main.741.

RUSSELL B. & PEARS D. (1918). *The philosophy of logical atomism*. La Salle : Open Court, 7th. print édition.

STACEY J., MINERVINI P., DUBOSSARSKY H., CAMBURU O.-M. & REI M. (2024). Atomic inference for NLI with generated facts as atoms. In Y. AL-ONAIZAN, M. BANSAL & Y.-N. CHEN, Éds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, p. 10188–10204, Miami, Florida, USA : Association for Computational Linguistics. DOI : 10.18653/v1/2024.emnlp-main.569.

WANG Z., GAO C., XIAO C., HUANG Y., SI S., LUO K., BAI Y., LI W., DUAN T., LV C., LU G., CHEN G., QI F. & SUN M. (2025). Document segmentation matters for retrieval-augmented generation. In W. CHE, J. NABENDE, E. SHUTOVA & M. T. PILEHVAR, Éds., *Findings of the Association for Computational Linguistics : ACL 2025*, p. 8063–8075, Vienna, Austria : Association for Computational Linguistics. DOI : 10.18653/v1/2025.findings-acl.422.

WANNER M., EBNER S., JIANG Z., DREDZE M. & VAN DURME B. (2024). A closer look at claim decomposition. In D. BOLLEGALA & V. SHWARTZ, Éds., *Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (*SEM 2024)*, p. 153–175, Mexico City, Mexico : Association for Computational Linguistics. DOI : 10.18653/v1/2024.starsem-1.13.

YOU X., LUO Y. & GU L. (2025). MS-RAG : Simple and effective multi-semantic retrieval-augmented generation. In C. CHRISTODOULOPOULOS, T. CHAKRABORTY, C. ROSE & V. PENG, Éds., *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, p. 22609–22625, Suzhou, China : Association for Computational Linguistics. DOI : 10.18653/v1/2025.emnlp-main.1151.

A Prompts utilisés

A.1 PROMPT 1 (Première passe)

```
You are an expert in disambiguation and information extraction.
You must decompose the text into atomic propositions (single facts) that
are FULLY AUTONOMOUS.
ABSOLUTE RULES:
1. ZERO PRONOUNS: "He", "She", "They", "His", "Her", "Its", "This one"
ARE FORBIDDEN. ALWAYS replace them with the full name of the entity.
2. CONTEXT: Each sentence must be readable alone without knowing its
source.
3. REPETITION: Repeat the subject in EACH sentence.
OUTPUT FORMAT: Only a JSON array of strings.
Title: title
Content: content
Output:
```

A.2 PROMPT 2 (Second Pass)

```
You are an expert in linguistics and formal logic. Your task is to verify
 and correct "Atomic Propositions".

Here are the strict rules a proposition must follow:

1. ATOMICITY (Structure)
- It must express a SINGLE fact.
- No conjunctions like "and", "but also" linking several distinct facts.
- No enumerations (e.g., "He likes X, Y, and Z").
- Allowed modifiers: Place, Time, Manner (only if they are inseparable
from the main fact).
- ATOMIC Example: "A healthy carrier is an individual."
- NON-ATOMIC Example: "Enclosures are churches surrounded by a wall." (2
facts: being a church + being surrounded).

2. AUTONOMY (Semantics)
- It must be COMPLETE and understandable without external context.
- NO ambiguous pronouns ("He", "She", "They" without a referent).
- NO deictics ("This", "That", "These" without a referent).
- All entities must be explicitly named.
```

```
- AUTONOMOUS Example: "The Anglo-Arab is a breed of horse."
- NON-AUTONOMOUS Example: "The cemetery is bounded." (Which cemetery?).

For each given proposition:
1. Check if it is VALID (respects both criteria).
2. If NO, propose a REFORMULATION that makes the proposition Atomic and
Autonomous.
   - If the sentence contains several facts -> Split it (you can return
multiple propositions).
   - If the sentence is "meta" or useless -> Mark it as DELETE.
   - If entities are missing -> Reintroduce the explicit subject (use the
 title of the provided document or the implicit context).

Expected JSON output format:
[
  {
    "original": "...",
    "status": "VALID" | "MODIFIED" | "SPLIT" | "DELETE",
    "correction": ["..."] (List of 1 or more corrected strings)
  },
  ...
]

Here is the subject of the document: "{subject}"

Here are the propositions to verify:
{propositions}

Use this context (original passage) ONLY to resolve ambiguities (pronouns
, missing entities) during reformulation:
---
{context}
---

Analyze them according to the criteria (Atomicity, Autonomy). Return the
JSON.
```

A.3 PROMPT 3 (ClosedIE)

```
Extract all factual (subject, predicate, object) triples from the
sentence.
One triple per line in the format:
subject | predicate | object
No explanations. If no triple can be extracted, write nothing.

Here the vocabulary of relations : {list_relations}

Sentence: {text}
```

A.4 PROMPT 4 (OpenIE)

```
Extract all factual (subject, predicate, object) triples from the
sentence.
One triple per line in the format:
subject | predicate | object
No explanations. If no triple can be extracted, write nothing.

Sentence: {text}
```

B Résultats complets

Cette annexe présente l'intégralité des résultats quantitatifs obtenus lors de nos différentes évaluations. Les tableaux 6, 7 et 8 détaillent respectivement les performances complètes de nos approches sur les corpus CaRB (extraction en vocabulaire ouvert), SMiLER (décliné par langue) et FewRel (extraction en vocabulaire fermé), ainsi que sur DocRED (extraction au niveau du document). Par ailleurs, concernant la phase d'apprentissage, nous observons dans la figure 1 que l'entraînement du modèle `MPropositionneur-V2` converge rapidement vers une solution stable autour de l'itération 2000.

	CoreNLP				Qwen3-0.6B				Qwen3-4B			
Pipeline	P	R	F1	AUC	P	R	F1	AUC	P	R	F1	AUC
Direct	**17.6**	24.3	20.4	0.144	**29.4**	7.9	12.4	0.051	**46.2**	33.3	**38.7**	0.244
Prop	16.4	**31.3**	**21.5**	**0.182**	24.0	11.1	15.2	0.069	31.6	35.0	33.2	0.230
Union	–	–	–	–	23.7	**14.7**	**18.1**	**0.091**	30.8	**43.6**	36.1	**0.285**

TABLE 6 – Résultats sur le jeu de données CaRB. L'atomisation améliore significativement le rappel et le F1 quand l'extracteur est basique. Cependant, avec une extraction par LLM, elle diminue.

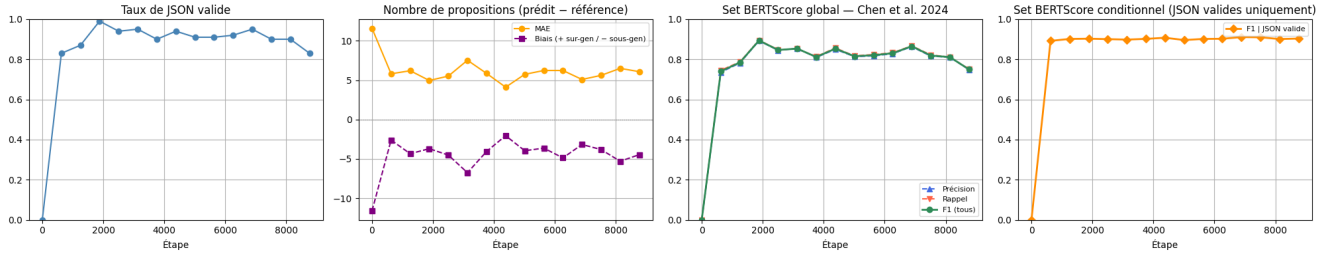

FIGURE 1 – Évaluation de `MPropositionneur-V2` (Qwen3-0.6B) durant son entraînement. Nous avons gardé le modèle qui maximise le F1 du score BERT ensembliste.

C Théorie de l'information sémantique

C.1 Définitions

Soit $\mathcal{W}$ l'ensemble des mondes possibles (structures interprétatives).

Définition 1 (Contenu sémantique)**.** *Soit ϕ une formule. Le contenu sémantique de ϕ est l'ensemble des mondes qui satisfont ϕ :*

$$\mathrm{Cont}(\phi) = \{w \in \mathcal{W} \mid w \models \phi\}$$

Benchmark	Pipeline	GLiREL			Qwen3-0.6B			Qwen3-4B		
		Acc	e-rec	r-rec	Acc	e-rec	r-rec	Acc	e-rec	r-rec
SMiLER	English – Direct	49.8	99.0	50.3	35.7	100.0	35.7	71.4	100.0	71.4
	English – Prop	43.7	86.6	50.4	35.1	86.6	40.5	59.6	86.6	68.8
	English – Comb	**51.5**†	**99.1**	**51.9**	–	–	–	–	–	–
	French – Direct	65.2	**99.8**	65.3	32.2	100.0	32.2	81.8	100.0	81.8
	French – Prop	48.9	76.7	63.8	39.7	76.7	51.7	64.6	76.7	84.3
	French – Comb	**66.3**	**99.8**	**66.4**	–	–	–	–	–	–
	German – Direct	59.2	99.6	59.4	22.9	100.0	22.9	76.5	100.0	76.5
	German – Prop	49.2	85.1	57.8	46.3	85.1	54.5	68.7	85.1	80.8
	German – Comb	**59.4**	**99.7**	**59.5**	–	–	–	–	–	–
	Spanish – Direct	**46.5**	**99.6**	46.7	24.8	100.0	24.8	65.0	100.0	65.0
	Spanish – Prop	38.5	79.2	**48.6**	27.4	79.2	34.6	58.9	79.2	74.3
	Spanish – Comb	**46.5**	**99.6**	46.7	–	–	–	–	–	–
	Portuguese – Direct	59.3	**99.4**	59.7	32.8	100.0	32.8	77.6	100.0	77.6
	Portuguese – Prop	56.4	87.7	64.3	43.2	87.7	49.2	71.9	87.7	82.0
	Portuguese – Comb	**63.2**†	**99.4**	**63.5**	–	–	–	–	–	–
	Italian – Direct	63.9	99.4	64.3	36.2	100.0	36.2	81.8	100.0	81.8
	Italian – Prop	54.1	82.2	65.8	41.8	82.2	50.8	69.5	82.2	84.5
	Italian – Comb	**67.7**†	**99.6**	**68.0**	–	–	–	–	–	–
	Macro-avg – Direct	57.3	**99.5**	57.6	30.8	**100.0**	30.8	**75.7**	**100.0**	75.7
	Macro-avg – Prop	48.5	82.9	58.5	**38.9**	82.9	**46.9**	65.5	82.9	**79.1**
	Macro-avg – Comb	**59.1**†	**99.5**	**59.3**	–	–	–	–	–	–
FewRel	Direct	48.7	**100.0**	48.7	**42.2**	**100.0**	42.2	**67.7**	**100.0**	67.7
	Prop	40.2	75.8	53.1	40.3	75.8	**53.2**	51.5	75.8	**68.0**
	Comb	**50.0**†	**100.0**	**50.0**	–	–	–	–	–	–

TABLE 7 – Résultats sur les jeux de données FewRel et SMiLER. † indique une amélioration statistiquement significative de Comb par rapport à Direct (test de bootstrap, $p < 0.05$). L'atomisation améliore significativement le rappel de relations. Cependant, le rappel d'entités diminue. La précision augmente lorsque nous passons à un petit LLM (Qwen3-0.6B).

Pipeline	GLiREL			Qwen3-0.6B			Qwen3-4B		
	P	R	F1	P	R	F1	P	R	F1
Sentence	3.7	9.4	5.3	22.6	8.3	12.1	**36.6**	13.3	19.5
Document	1.4	14.8	2.5	20.3	**15.7**	**17.7**	32.0	**24.4**	**27.7**
Proposition	**5.9**	6.9	**6.3**	**25.9**	6.7	10.6	36.5	9.4	14.9
Sentence+Proposition	3.8	12.3	5.8	–	–	–	–	–	–
Document+Proposition	1.6	**18.1**	2.9	–	–	–	–	–	–

TABLE 8 – Résultats sur le jeu de données DocRED. L'atomisation améliore significativement la précision et le F1 lors de l'utilisation d'un extracteur plus faible. Cependant, avec une extraction par LLM, elle diminue.

Définition 2 (Information sémantique (Bar-Hillel & Carnap, 1953))**.** *Pour toute formule close ϕ, le contenu informationnel $I(\phi)$ est défini par :*

$$I(\phi) = -\log_2\left(\frac{|\mathrm{Cont}(\phi)|}{|\mathcal{W}|}\right) = -\log_2(\mu(\mathrm{Cont}(\phi)))$$

Intuition. Plus une formule est spécifique (peu de mondes la satisfont), plus elle contient d'information. Une tautologie ($\mathrm{Cont}(\phi) = \mathcal{W}$) a une information nulle, tandis qu'une contradiction ($\mathrm{Cont}(\phi) = \emptyset$) a une information infinie.

Définition 3 (Proposition atomique)**.** *Une proposition est dite **atomique** si elle est une formule close (*AUTONOMIE*) et qu'on ne peut pas la décomposer plus sans faire une mauvaise décomposition*

*(*ATOMICITÉ*).*

C.2 Exemples

Information de $A \wedge B$. Soient A et B deux variables propositionnelles indépendantes. Il y a quatre états possibles ($|\mathcal{W}| = 4$) et la conjonction n'est vraie que dans un seul monde ($|\mathrm{Cont}(A \wedge B)| = 1$) :

$$I(A \wedge B) = -\log_2\left(\tfrac{1}{4}\right) = 2 \text{ bits.}$$

Information de $A \vee B$. La disjonction est vraie dans trois mondes ($|\mathrm{Cont}(A \vee B)| = 3$) :

$$I(A \vee B) = -\log_2\left(\tfrac{3}{4}\right) \approx 0{,}415 \text{ bits.}$$

Information de A. La variable A seule est vraie dans deux mondes ($|\mathrm{Cont}(A)| = 2$) :

$$I(A) = -\log_2\left(\tfrac{2}{4}\right) = 1 \text{ bit.}$$

On observe ainsi la hiérarchie $I(A \wedge B) > I(A) > I(A \vee B)$, cohérente avec l'idée que la conjonction est plus informative que chacun de ses membres, et la disjonction moins informative.

C.3 Hypothèse de travail

Hypothèse 1 (Non-trivialité)**.** *Nous considérons des propositions contingentes et logiquement indépendantes, ce qui assure $\mu(\mathrm{Cont}(\phi)) \in]0, 1[$.*

C.4 Décomposition

Définition 4 (Décomposition sûre)**.** *Une décomposition d'une formule ϕ en une formule ψ est dite **sûre** (resp. **strictement sûre**) si ψ est une sous-formule de ϕ et que $I(\phi) \geq I(\psi)$ (resp. $I(\phi) > I(\psi)$).*

Au contraire, une décomposition sera dite « mauvaise » si elle augmente ou conserve l'information :

Définition 5 (Mauvaise décomposition)**.** *La décomposition d'une formule ϕ en une formule ψ est dite **mauvaise** si ψ est une sous-formule stricte de ϕ et que $I(\phi) \leq I(\psi)$.*

C.5 Preuves des lemmes et du théorème

Lemme 1 (Divisibilité de la conjonction)**.** *Soit $\phi = A \wedge B$ où A et B sont logiquement indépendants. L'extraction de la composante A est une opération **strictement sûre**.*

Démonstration. Par définition de la conjonction, $\mathrm{Cont}(A \wedge B) = \mathrm{Cont}(A) \cap \mathrm{Cont}(B)$. Puisque A et B sont indépendants, $\mathrm{Cont}(A \wedge B) \subsetneq \mathrm{Cont}(A)$. Par monotonicité de μ, on a $\mu(\mathrm{Cont}(A \wedge B)) < \mu(\mathrm{Cont}(A))$. La fonction $-\log_2$ étant strictement décroissante, il vient :

$$I(A \wedge B) > I(A).$$

L'information de A est strictement inférieure à celle de $A \wedge B$: l'opération est strictement sûre. □

Lemme 2 (Indivisibilité de la disjonction)**.** *Soit $\phi = A \vee B$ où A et B sont logiquement indépendants. L'extraction de la composante A est une opération* ***mauvaise****.*

Démonstration. Par définition de la disjonction, $\mathrm{Cont}(A \vee B) = \mathrm{Cont}(A) \cup \mathrm{Cont}(B)$. On a l'inclusion stricte $\mathrm{Cont}(A) \subsetneq \mathrm{Cont}(A \vee B)$, donc $\mu(\mathrm{Cont}(A)) < \mu(\mathrm{Cont}(A \vee B))$, d'où :

$$I(A) > I(A \vee B).$$

L'information de A est strictement supérieure à celle de $A \vee B$: décomposer une disjonction hallucine un fait plus précis que ce qui était affirmé. □

Lemme 3 (Indivisibilité de l'implication)**.** *Soit $\phi = A \to B$ où A et B sont logiquement indépendants. L'extraction de la composante A est une opération* ***mauvaise****.*

Démonstration. On a $\mathrm{Cont}(A \to B) = \mathrm{Cont}(\neg A) \cup \mathrm{Cont}(B)$. Puisque A et B sont indépendants, $\mathrm{Cont}(A) \subsetneq \mathrm{Cont}(\neg A) \cup \mathrm{Cont}(B) = \mathrm{Cont}(A \to B)$ est impossible en général. Calculons directement : sous l'hypothèse d'indépendance, $\mu(\mathrm{Cont}(A \to B)) = 1 - \mu(\mathrm{Cont}(A)) \cdot \mu(\mathrm{Cont}(\neg B))$. Par non-trivialité, $\mu(\mathrm{Cont}(A)), \mu(\mathrm{Cont}(\neg B)) \in]0, 1[$, donc $\mu(\mathrm{Cont}(A \to B)) > \mu(\mathrm{Cont}(A))$, d'où :

$$I(A) > I(A \to B).$$

L'information de A est strictement supérieure à celle de $A \to B$: la décomposition est mauvaise. □

Théorème 1 (Caractérisation structurelle)**.** *Une formule ϕ est* ***atomique*** *si et seulement si elle est logiquement équivalente à une* ***clause*** *(disjonction finie de littéraux).*

Démonstration. Soit ϕ mise sous Forme Normale Conjonctive (FNC) :

$$\phi \equiv C_1 \wedge C_2 \wedge \cdots \wedge C_n,$$

où chaque C_i est une clause (disjonction de littéraux).

($\Rightarrow$) Supposons ϕ atomique. Par l'absurde, supposons $n \geq 2$ avec une FNC irrédondante (aucune clause n'est impliquée par les autres). Les clauses C_1 et $C_2 \wedge \cdots \wedge C_n$ sont alors logiquement indépendantes, et d'après le lemme de divisibilité, l'opération $\phi \mapsto C_1$ est strictement sûre, ce qui contredit l'atomicité. Donc $n = 1$: ϕ est une unique clause.

($\Leftarrow$) Supposons $\phi = L_1 \vee \cdots \vee L_k$. Pour toute sous-disjonction stricte $\psi = L_{i_1} \vee \cdots \vee L_{i_m}$ avec $m < k$, on a $\mathrm{Cont}(\psi) \subsetneq \mathrm{Cont}(\phi)$ (par indépendance des littéraux), donc $\mu(\mathrm{Cont}(\psi)) < \mu(\mathrm{Cont}(\phi))$ et $I(\psi) > I(\phi)$. Toute tentative de division est donc mauvaise, et ϕ atomique. □