Les méthodes d'idéation structurées aident-elles encore les modèles de pointe ? Une étude de redécouverte de brevets en aveugle

Alaric Tabariès

18 novembre 2026 · Carnet d'Alaric Tabariès
altab.fr/fr/billets/ideation-structuree-redecouverte-brevets/

Résumé

De nombreux outils récents enrobent un grand modèle de langage d'une méthode d'idéation structurée (TRIZ, catalogues d'effets, analogie inter-domaines), en partant du principe que guider le modèle donne de meilleures solutions que de lui demander simplement des idées. Nous testons ce principe par un protocole de redécouverte en aveugle. À partir de 24 demandes de brevets européennes et internationales publiées après les dates limites de connaissances déclarées des modèles testés, nous rédigeons des énoncés de problème à partir de la seule section d'art antérieur, puis nous vérifions si chaque approche produit une idée correspondant au mécanisme de la revendication principale. Nous comparons le prompt simple (7 ou 30 idées) à une procédure TRIZ par matrice des contradictions, à un passage forcé sur un catalogue de 255 effets physiques et à une analogie fonctionnelle avec recherche de brevets, avec Claude Fable 5.2 et GPT-6.1 Sol comme solveurs et comme juges croisés. Aucune méthode structurée ne retrouve plus d'inventions que le prompt simple, quelle que soit la configuration. Avec GPT-6.1 Sol, deux exécutions identiques à 30 idées retrouvent 5 et 8 inventions sur 24, tandis que le catalogue et TRIZ en retrouvent 6 et 8 : l'écart entre deux exécutions du prompt simple est aussi grand que l'écart entre le prompt simple et n'importe quelle méthode. Une complémentarité apparente du catalogue d'effets avec un modèle ne se reproduit pas avec l'autre. Nous montrons aussi que des énoncés rédigés avec accès au brevet complet gonflent les taux de redécouverte (17 sur 24 contre 8 sur 24 pour le même modèle). L'étude est exploratoire : petit échantillon, juges LLM. Nous publions le protocole, les énoncés, les listes d'idées et les verdicts.

Mots-clés. grands modèles de langage · TRIZ · idéation · brevets · évaluation

---

Traduction française de la version anglaise, qui fait foi.

1. Introduction

Les méthodes d'idéation structurées ont été conçues pour écarter les ingénieurs humains de leurs solutions habituelles. TRIZ associe une contradiction technique à des principes inventifs ; les catalogues d'effets listent des phénomènes physiques à envisager ; l'analogie de conception explore des domaines éloignés à la recherche d'un mécanisme qui remplit la même fonction. Un nombre croissant d'outils applique aujourd'hui ces méthodes à des grands modèles de langage, en faisant parcourir au modèle les étapes de la méthode par des prompts.

Ces outils partagent une hypothèse rarement testée : la méthode améliore les idées que le modèle aurait produites si on le lui avait simplement demandé. Les articles qui présentent ces outils rapportent en général des études de cas ou la fidélité à la méthode, sans comparaison contrôlée avec le même modèle sollicité simplement. Des comparaisons plus larges de stratégies de prompt existent, mais elles mesurent la divergence par des distances d'embeddings entre les idées générées, sans référence externe pour juger de leur utilité.

Nous posons une question plus étroite et plus difficile. Face à un vrai problème d'ingénierie, un modèle retrouve-t-il l'invention qui a réellement été brevetée pour y répondre, et une méthode structurée rend-elle cela plus probable ? La redécouverte d'une invention tenue à l'écart donne

 CC BY 4.0

une vérité de terrain qui ne dépend pas du goût d'un juge pour la nouveauté. Pour garder cette référence hors des connaissances des modèles, nous utilisons des demandes de brevets publiées après leur date limite de connaissances déclarée.

Nos contributions sont les suivantes :

1. Un protocole de redécouverte de brevets en aveugle pour l'idéation en ingénierie, avec des classes de brevets tirées au hasard, des énoncés écrits à partir de l'art antérieur seul, un audit explicite des fuites, des listes anonymisées et deux juges.
2. Une comparaison de trois familles de méthodes avec le prompt simple, à longueur de liste égale, sur deux modèles de pointe, chaque modèle jugeant aussi les listes de l'autre.
3. La preuve que la variation d'une exécution à l'autre du prompt simple est aussi grande que n'importe quelle différence entre le prompt simple et une méthode, ce qu'une référence à exécution unique masquerait.
4. Une mesure de l'ampleur avec laquelle les fuites dans l'énoncé du problème gonflent les taux de redécouverte.

2. Travaux connexes

Outils à base de LLM fondés sur des méthodes d'idéation. AutoTRIZ (Jiang et al., 2024) automatise le flux de travail TRIZ des contradictions ; il est évalué sur la détection de contradictions face à des cas de manuel et sur une étude de cas, et ses bénéfices annoncés sont l'accessibilité et un processus interprétable. TRIZ Agents (Szczepanik et Chudziak, 2025) répartit ce flux de travail entre plusieurs agents et compare chaque étape d'une étude de cas d'ingénierie à l'étude humaine d'origine. Aucun des deux ne rapporte de comparaison contrôlée de la qualité des solutions avec le même modèle sollicité sans la méthode. Une étude en science des matériaux moléculaires (Gusarov et al., 2026) fait tourner un mode de référence sans contraintes TRIZ à côté de son flux de travail guidé par TRIZ, un modèle évaluateur notant les deux ; sur neuf modèles locaux et onze problèmes de chimie, les scores de qualité et de nouveauté des deux modes restent dans la variation entre modèles, et le bénéfice rapporté est une moindre consommation de tokens. Elle n'utilise aucune solution de référence tenue à l'écart. Ces outils prolongent une longue lignée de travaux sur TRIZ (Altshuller, 1984) et sur l'analogie computationnelle (Hope et al., 2017; Kang et al., 2022).

Prompts de méthode contre prompts simples. IDEAFix (Carichon et al., 2026) est le travail le plus proche. Il compare 25 prompts issus de huit familles, dont TRIZ, SCAMPER, la théorie C-K et le design thinking, à des prompts de contrôle, sur cinq modèles et 81 énoncés. Il rapporte que des prompts simples assortis de consignes explicites de divergence égalent ou battent souvent les prompts fondés sur une méthode. Ses mesures sont des distances d'embeddings automatiques (diversité, nouveauté par rapport aux sorties de contrôle, rareté) ; il n'emploie ni évaluateurs humains ni vérité de terrain externe, implémente chaque méthode sous la forme d'un seul prompt et n'analyse pas les gains par modèle. Ses auteurs appellent à des mesures d'utilité et à des procédures itératives en plusieurs étapes. Notre étude répond à ces deux points avec une autre mesure, et ajoute l'analogie par recherche documentaire et la réplication entre modèles. Meincke et al. (2024) comparent 35 stratégies de prompt pour la diversité des idées.

Rendements décroissants des techniques de prompt. Des études en génie logiciel (Wang et al., 2024) et en questions-réponses (Meincke et al., 2025) constatent que les techniques développées pour des modèles plus anciens aident moins, voire nuisent, avec les modèles récents. Nous ne connaissons pas d'étude équivalente pour l'idéation en ingénierie.

La redécouverte comme évaluation. Plusieurs bancs d'essai demandent aux modèles de retrouver une idée ou un résultat de recherche publié, sous une date limite temporelle, parmi lesquels Reconstruction (Chen et al., 2026), MOOSE-Chem (Yang et al., 2024), ResearchBench (Liu et al., 2025), AI Idea Bench 2025 (Qiu et al., 2025) et FIRE-Bench (Wang et al., 2026). Tous

visent la recherche scientifique. Nous n'en avons pas trouvé qui prenne des inventions brevetées pour référence.

Les LLM comme juges d'idées. Les jugements de nouveauté par des LLM sont sensibles à la formulation du prompt et peuvent diverger de ceux des experts (Ji et al., 2026; Si et al., 2024). Nous évitons donc les notes de nouveauté : les juges décident si une idée divulgue le mécanisme d'une revendication précise, une tâche plus étroite, et nous vérifions la stabilité par un nouveau jugement en aveugle et par un juge d'un autre fournisseur.

3. Méthode

Le protocole a été écrit avant toute exécution ; les modifications ultérieures sont datées dans le fichier du protocole.

3.1 Échantillon de brevets

Nous avons tiré au hasard 36 classes CPC, après avoir retiré neuf classes : H10 (dispositifs semi-conducteurs), C06 (explosifs), F22 et F23 (production de vapeur, combustion), C07 et C12 (chimie organique, biochimie), G06 et H04 (informatique, communication) et A61 (médecine). Pour chaque classe, un agent de préparation a interrogé les EPO Open Patent Services pour les demandes EP et WO publiées à partir du 1er août 2026, a examiné les résultats dans l'ordre renvoyé et a retenu le premier qui remplissait des critères fixes : la revendication 1 décrit un mécanisme physique ou chimique identifiable ; aucun membre de la famille n'a été publié avant le 1er août 2026 ; le texte des revendications est disponible. 24 demandes ont été retenues, une par classe, parmi 158 candidats consignés ; chaque rejet est consigné avec son motif. Comme le service renvoie d'abord les publications les plus récentes, les demandes retenues (9 EP, 15 WO) ont toutes été publiées entre le 29 octobre et le 11 novembre 2026, avec des dates de priorité en avril et mai 2025.

Modèle	Date limite de connaissances déclarée	Rôle
Claude Fable 5.2	fin juin 2026	solveur, juge, préparation
GPT-6.1 Sol	30 avril 2026	solveur et juge, seconde série
GPT-5.6 Sol	16 février 2026	solveur et juge, première série

3.2 Énoncés de problème

Première série (v1). Un agent de préparation a rédigé chaque énoncé à partir du brevet complet, avec la consigne de ne pas révéler la solution.

Seconde série (v2). Un agent a extrait uniquement la section de contexte et d'art antérieur de la description, en s'arrêtant avant toute phrase annonçant l'invention. Un second agent, qui n'a jamais vu les revendications, a rédigé l'énoncé à partir de ce seul extrait.

Audit des fuites. Un agent distinct a comparé chaque énoncé à son brevet et évalué la fuite : aucune, orientation (les contraintes ou objectifs ne laissent guère place qu'à la famille de solutions) ou fuite manifeste. La même grille a été appliquée aux deux séries.

3.3 Approches

Chaque paire (problème, approche) a été exécutée dans une session neuve, sans accès au web ni aux autres sorties.

Code	Approche	Budget
T7	Prompt simple, 7 meilleures idées	7
T30	Prompt simple, 30 idées variées ; exécuté deux fois en v2	30
AT	TRIZ par matrice des contradictions, d'après AutoTRIZ : énoncer une contradiction parmi les 39 paramètres, chercher les principes dans la matrice	30

 CC BY 4.0

Code	Approche	Budget
	classique par un script, écrire des solutions pour chaque principe renvoyé ; passes répétées	
CA	Passage forcé sur un catalogue de 255 effets physiques, construit par un agent sans connaissance des cas ; idées issues uniquement des entrées retenues	30
FA	Analogie fonctionnelle avec recherche : abstraire le problème en fonctions, chercher dans des domaines éloignés des brevets et articles antérieurs à la date de priorité, transposer	30

Toutes les sorties ont le même format (titre, mécanisme, mise en œuvre) et ne portent aucune trace de la méthode, de sorte que les juges ne peuvent pas distinguer les listes. FA n'a été exécutée en v1 qu'avec Claude.

3.4 Jugement

Les listes ont été anonymisées par des étiquettes aléatoires pour chaque cas. Deux juges indépendants ont noté chaque idée par rapport à la référence : 2 pour le même mécanisme que la revendication 1, appliqué au même endroit du système ; 1 pour une parenté partielle ; 0 sinon ; les égalités étant tranchées à la baisse. Une invention est considérée comme retrouvée lorsque les deux juges donnent la note 2 à au moins une idée. Contrôles : des listes déjà jugées ont été mêlées en aveugle à des lots ultérieurs ; un troisième juge a appliqué une grille reformulée à un sous-ensemble ; et chaque modèle a jugé les listes produites par l'autre.

3.5 Exécution

Les exécutions de Claude ont utilisé des sous-agents dans un espace de travail isolé. Les exécutions de GPT ont utilisé la CLI Codex en mode non interactif, une session neuve par paire, avec la recherche web désactivée et l'accès réseau des commandes bloqué, à partir des mêmes fichiers d'énoncé, de consignes, de matrice et de catalogue.

4. Résultats

Aucune méthode structurée ne retrouve plus d'inventions que le prompt simple, quelle que soit la configuration, et des exécutions répétées du prompt simple diffèrent autant que les méthodes entre elles. Les comptes sont des inventions retrouvées selon les deux juges.

4.1 Fuites dans les énoncés

Niveau de fuite	Énoncés v1	Énoncés v2
Aucune	3	12
Orientation	19	12
Fuite manifeste	2	0

Avec le même modèle et le même prompt à 7 idées, Claude retrouve 17 inventions sur 24 à partir des énoncés v1 et 8 sur 24 à partir des énoncés v2. Trois énoncés v2 ont été jugés trop vagues pour être résolubles, parce que le contexte du brevet tient en quelques phrases.

4.2 Première série (énoncés v1, 24 cas)

Approche	Claude Fable 5.2	GPT-5.6 Sol
T7	17	13
T30	18	15
AT	16	10
CA	17	12

 CC BY 4.0

Appariée à T30 par un test exact de McNemar (McNemar, 1947), AT gagne 1 cas et en perd 3 avec Claude (p = 0,63) et en gagne 2 et en perd 7 avec GPT-5.6 Sol (p = 0,18) ; sur les deux modèles réunis, 3 gains et 10 pertes (p = 0,09). CA gagne 1 cas et en perd 2 avec Claude, en gagne 2 et en perd 5 avec GPT-5.6 Sol. Les méthodes repoussent aussi l'idée gagnante vers le bas de la liste : elle se trouve parmi les sept premières idées dans 16 des listes T30 mais dans 8 des listes AT avec Claude.

FA, exécutée avec Claude sur 23 cas, retrouve 14 inventions contre 17 pour T30 (1 gain, 4 pertes, p = 0,38). Sa consigne n'autorisait que des idées rattachées à une source retrouvée, et la recherche de brevets portait sur les résumés, sous de fortes limitations de débit.

Les deux modèles retrouvent des inventions différentes : toutes approches confondues, 22 inventions sur 24 sont retrouvées par au moins l'un des deux.

4.3 Seconde série (énoncés v2)

GPT-6.1 Sol, les 24 cas.

Approche	Retrouvées sur 24
T7	5
T30, première exécution	5
T30, seconde exécution	8
CA	6
AT	8

Claude Fable 5.2. T7 retrouve 8 inventions sur 24. Sur les 16 cas qu'elle manque :

Approche	Retrouvées sur 16
T30, première exécution	3
T30, seconde exécution	3
AT	3
CA	4

Ce que chaque ajout apporte à une première exécution de T30.

Ajouté à T30	Nouveaux cas, Claude	Nouveaux cas, GPT-6.1 Sol
Une seconde exécution de T30	1	3
AT	1	4
CA	3	2

Le catalogue semble complémentaire avec Claude et non avec GPT-6.1 Sol ; TRIZ semble complémentaire avec GPT-6.1 Sol et non avec Claude. Aucun de ces schémas ne se vérifie d'un modèle à l'autre, et les deux restent dans l'écart entre deux exécutions identiques du prompt simple.

4.4 Fiabilité des juges

Contrôle	Accord sur « retrouvée »
Deux juges Claude, listes Claude v1	98 %, kappa de Cohen 0,95 (Cohen, 1960)
Deux juges Claude, listes Claude v2	64 sur 64
Deux juges Claude, listes GPT v2	110 sur 120
Nouveau jugement en aveugle des listes par de nouveaux juges	36 sur 36 identiques
Troisième juge avec grille reformulée	24 sur 26
Juges GPT-5.6 Sol contre juges Claude, listes Claude v1	92 sur 99, kappa 0,83
Juges GPT-6.1 Sol contre juges Claude, listes GPT v1	90 sur 96, kappa 0,88

 CC BY 4.0

Le classement des approches est le même selon les jugements de l'un ou l'autre modèle. L'ampleur du déficit de TRIZ dans la première série dépend du juge : 5 cas en dessous de T30 selon les juges Claude, 2 selon ceux de GPT.

5. Discussion

Le gain supposé n'est pas observé. Sur deux versions d'énoncés, trois familles de méthodes, deux modèles solveurs et deux modèles juges, aucune méthode n'a retrouvé plus d'inventions que le prompt simple à longueur de liste égale. Cela concorde avec IDEAFix (Carichon et al., 2026), qui constate sur des mesures de divergence que des prompts simples égalent souvent les prompts de méthode, et avec les rapports selon lesquels les techniques de prompt perdent de leur valeur sur les modèles récents (Meincke et al., 2025; Wang et al., 2024). Cela les prolonge à une mesure à vérité de terrain et à des procédures en plusieurs étapes.

La variance est la référence manquante. Deux exécutions identiques à 30 idées du même modèle sur les mêmes énoncés diffèrent de trois inventions retrouvées sur 24. Une méthode évaluée face à une seule exécution du prompt simple peut gagner ou perdre autant par simple hasard. Le contrôle approprié est le prompt simple répété avec le même nombre d'appels et d'idées que la méthode en consomme. Nos propres données illustrent le risque : les trois cas supplémentaires du catalogue avec Claude ressemblaient à de la complémentarité jusqu'à ce que l'autre modèle montre le schéma inverse.

Pourquoi les méthodes pourraient ne pas aider. Une matrice des contradictions ou un catalogue d'effets apporte des connaissances qu'un modèle de pointe possède déjà ; la procédure ne fait alors surtout que contraindre la partie de ces connaissances qui est utilisée. Quand le principe ou l'entrée imposés n'ont aucune prise sur le problème, la liste se remplit d'idées forcées et l'idée gagnante descend. La méthode par recherche documentaire a remplacé les idées propres du modèle par celles qu'elle pouvait rattacher à une source, et a perdu des cas que le modèle trouvait seul. Un montage qui ajoute les idées retrouvées à celles du modèle, au lieu de les y substituer, n'a pas été testé.

Diversité des modèles contre diversité des méthodes. Dans la première série, les deux modèles ensemble ont retrouvé 22 inventions sur 24, plus que n'importe quelle approche ajoutée à l'un ou l'autre. Changer de modèle a apporté plus de nouvelles bonnes idées que changer de méthode.

Ce que cela ne montre pas. Les méthodes ont été conçues pour des personnes. Savoir si TRIZ ou la recherche d'analogies aident des ingénieurs humains, seuls ou avec un modèle, sort du cadre de cette étude, tout comme l'éventuel bénéfice de traçabilité ou de processus partagé qu'apporte un rapport structuré. La redécouverte ignore aussi les bonnes idées qui diffèrent de celle qui a été brevetée. Enfin, l'absence d'avantage observé sur un petit échantillon n'est pas une preuve d'équivalence.

Où la valeur pourrait se trouver. Les résultats éloignent du guidage procédural et orientent vers ce qu'un outil apporte et que le modèle n'a pas : des documents postérieurs à sa date limite, des données privées, les contraintes non dites d'un expert, et la vérification des idées par rapport à l'art antérieur. Ces pistes n'ont pas été évaluées ici.

6. Limites

- **Taille de l'échantillon.** 24 brevets et une ou deux exécutions par approche. Aucune différence n'est statistiquement significative, et seuls de grands effets étaient détectables.
- **Fuite résiduelle.** La moitié des énoncés v2 orientent encore vers la famille de solutions, car ils dérivent de la façon dont le brevet lui-même cadre le problème.

 CC BY 4.0

- **Divulgation antérieure.** Nous avons vérifié les familles de brevets, pas les articles, produits ou exposés antérieurs des inventeurs. « Publié après la date limite déclarée » ne garantit pas que l'idée était inconnue. Les faibles taux de redécouverte en v2 et les cas différents retrouvés par chaque modèle ne suggèrent pas de mémorisation.
- **Juges LLM, pas d'évaluateurs humains.** L'accord est élevé et stable, mais un cas montre un désaccord systématique entre juges, et aucun expert n'a encore audité un échantillon.
- **Fidélité des méthodes.** AT suit le flux de travail publié d'AutoTRIZ avec une copie publique de la matrice classique, et non le code des auteurs, et est étendue à 30 idées par passes répétées. FA est notre propre implémentation, exécutée sous des limitations de débit de recherche.
- **Versions de modèles.** Les exécutions GPT ont utilisé GPT-5.6 Sol dans la première série et GPT-6.1 Sol dans la seconde. FA n'a été exécutée qu'avec un modèle et une version d'énoncés.
- **Même famille de modèles pour la préparation, la résolution et le jugement** dans le bras Claude, atténué par le jugement croisé entre modèles.
- **Base d'échantillonnage.** Demandes EP et WO dont le texte est disponible, toutes publiées dans une fenêtre de deux semaines ; neuf classes techniques exclues.

Déclaration d'assistance par IA

L'étude a été conçue et menée avec Claude (Anthropic) comme assistant de recherche sous la direction de l'auteur : rédaction du protocole, préparation des énoncés, exécutions de solveurs et de juges, analyse et premier jet de ce texte. Les exécutions GPT ont été lancées par l'auteur avec la CLI Codex. L'auteur est responsable du contenu.

Références

Altshuller, G. S. (1984). *Creativity as an exact science: The theory of the solution of inventive problems.* Gordon and Breach.

Carichon, F., Sharma, S., Girard, M., Rampa, R., & Farnadi, G. (2026). *IDEAFix: Evaluation framework for creative defixation prompting in LLMs* (arXiv:2606.00875). arXiv. https://arxiv.org/abs/2606.00875

Chen, S., Fei, Y., Liu, N., Yu, X., Li, L., Thapa, R., Ciobanu, M., Singh, N. P., Mao, Q., & Das, R. (2026). *Reconstruction: A blind benchmark for recovering research ideas from pre-publication bibliographies* (arXiv:2608.16645). arXiv. https://arxiv.org/abs/2608.16645

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement, 20*(1), 37–46. https://doi.org/10.1177/001316446002000104

Gusarov, S., Sapelnikova, S., Valdes, J. J., Hu, A., & Stoyanov, S. R. (2026). Integrating the theory of inventive problem solving with large language models: Enhancing reasoning for innovation in materials science at the molecular scale. *ChemEngineering, 10*(4), Article 54. https://doi.org/10.3390/chemengineering10040054

Hope, T., Chan, J., Kittur, A., & Shahaf, D. (2017). Accelerating innovation through analogy mining. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 235–243). ACM. https://doi.org/10.1145/3097983.3098038

Ji, F., Li, Y., Yang, J., Wu, J., Zhang, F., Cui, Z., Xie, Y., Peng, M., Xie, Q., Chen, X., & Xie, Z. (2026). *Style wins, substance loses: A diagnosis of LLM-as-judge in idea generation* (arXiv:2608.01666). arXiv. https://arxiv.org/abs/2608.01666

Jiang, S., Li, W., Qian, Y., Zhang, Y., & Luo, J. (2024). *AutoTRIZ: Automating engineering innovation with TRIZ and large language models* (arXiv:2403.13002). arXiv. https://arxiv.org/abs/2403.13002

Kang, H. B., Qian, X., Hope, T., Shahaf, D., Chan, J., & Kittur, A. (2022). Augmenting scientific creativity with an analogical search engine. *ACM Transactions on Computer-Human Interaction, 29*(6), Article 57. https://arxiv.org/abs/2205.15476

Liu, Y., Yang, Z., Xie, T., Ni, J., Gao, B., Li, Y., Tang, S., Ouyang, W., Cambria, E., & Zhou, D. (2025). *ResearchBench: Benchmarking LLMs in scientific discovery via inspiration-based task decomposition* (arXiv:2503.21248). arXiv. https://arxiv.org/abs/2503.21248

McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika, 12*(2), 153–157. https://doi.org/10.1007/BF02295996

Meincke, L., Mollick, E. R., Mollick, L., & Shapiro, D. (2025). *Prompting science report 2: The decreasing value of chain of thought in prompting.* Wharton Generative AI Labs. https://gail.wharton.upenn.edu/research-and-insights/tech-report-chain-of-thought/

Meincke, L., Mollick, E. R., & Terwiesch, C. (2024). *Prompting diverse ideas: Increasing AI idea variance* (arXiv:2402.01727). arXiv. https://arxiv.org/abs/2402.01727

Qiu, Y., Zhang, H., Xu, Z., Li, M., Song, D., Wang, Z., & Zhang, K. (2025). *AI Idea Bench 2025: AI research idea generation benchmark* (arXiv:2504.14191). arXiv. https://arxiv.org/abs/2504.14191

Si, C., Yang, D., & Hashimoto, T. (2024). *Can LLMs generate novel research ideas? A large-scale human study with 100+ NLP researchers* (arXiv:2409.04109). arXiv. https://arxiv.org/abs/2409.04109

Szczepanik, K., & Chudziak, J. A. (2025). TRIZ Agents: A multi-agent LLM approach for TRIZ-based innovation. In *Proceedings of the 17th International Conference on Agents and Artificial Intelligence* (Vol. 1, pp. 196–207). SCITEPRESS. https://doi.org/10.5220/0013321900003890

Wang, G., Sun, Z., Gong, Z., Ye, S., Chen, Y., Zhao, Y., Liang, Q., & Hao, D. (2024). *Do advanced language models eliminate the need for prompt engineering in software engineering?* (arXiv:2411.02093). arXiv. https://arxiv.org/abs/2411.02093

Wang, Z., Bai, F., Luo, Z., Su, J., Sun, K., Yu, X., Liu, J., Zhou, K., Cardie, C., Dredze, M., Hu, Z., & Xing, E. P. (2026). *FIRE-Bench: Evaluating AI agents on the rediscovery of scientific insights* (arXiv:2602.02905). arXiv. https://arxiv.org/abs/2602.02905

Yang, Z., Liu, W., Gao, B., Xie, T., Li, Y., Ouyang, W., Poria, S., Cambria, E., & Zhou, D. (2024). *MOOSE-Chem: Large language models for rediscovering unseen chemistry scientific hypotheses* (arXiv:2410.07076). arXiv. https://arxiv.org/abs/2410.07076

 CC BY 4.0