

NetSuite N/LLM : Comment classer automatiquement les motifs de remboursement

By houseblend.io Publié le 29 décembre 2025 32 min de lecture



Résumé Exécutif

Les entreprises modernes sont confrontées à un besoin urgent de **standardiser et d'analyser les motifs de remboursement** sur l'ensemble des canaux de vente. Les données non structurées ou incohérentes relatives aux « motifs de remboursement » – souvent du texte libre saisi par les clients ou les agents de support – entravent l'analyse, le recouvrement des revenus et la détection des fraudes. Historiquement, les organisations s'appuyaient sur le codage manuel ou sur des listes fixes de codes de motif, qui ne parviennent pas à saisir les explications nuancées ni à s'adapter au volume de données. En revanche, le **nouveau module N/LLM d'Oracle NetSuite** (une [interface SuiteScript](#) vers l'IA Générative d'OCI) permet l'*auto-classification* des motifs de remboursement à grande échelle. En tirant parti des grands modèles de langage (LLM) et des *embeddings* sémantiques au sein de NetSuite, les entreprises peuvent attribuer automatiquement les commentaires de remboursement en texte libre à des catégories standardisées. Ce rapport examine comment N/LLM peut être appliqué aux données des motifs de remboursement pour garantir la cohérence, améliorer les analyses et réduire la charge de travail manuelle. Nous passons en revue le contexte commercial des retours, examinons les approches techniques (traditionnelles vs. basées sur l'IA) et étudions les capacités de N/LLM (génération de texte, *embeddings*, RAG) pertinentes pour la classification. Les données provenant de sources industrielles soulignent les enjeux (par exemple, les volumes de retours du commerce électronique américain et les causes courantes de retour (Source: [www.glencoyne.com](#)) (Source: [corp.narvar.com](#)). Nous discutons d'un pipeline prototype : extraction du texte du motif de remboursement via [SuiteQL](#)/recherches enregistrées, génération d'*embeddings* ou utilisation d'invites LLM, et étiquetage automatique de chaque entrée. Des exemples de cas illustrent les avantages et les défis. Nous concluons par les implications pour la gouvernance, la précision et les futures orientations de l'IA de NetSuite.

Introduction et Contexte

Les remboursements et les retours sont un centre de coûts permanent dans le commerce de détail et le commerce électronique. En 2024, les retours du commerce de détail aux États-Unis ont été estimés à **890 milliards de dollars** (Source: www.glencoyne.com), un chiffre qui comprend les revenus perdus, les coûts de traitement et les amortissements d'inventaire. Les taux de retour en ligne typiques varient de **15 à 30 %** des commandes (jusqu'à 40 % dans l'habillement) (Source: www.glencoyne.com), ce qui signifie que **près d'un achat sur cinq** pourrait éventuellement être retourné. Les

motifs courants incluent « l'article n'allait pas », « non conforme à la description », « endommagé », « arrivé en retard », ou même « j'ai changé d'avis » (Source: corp.narvar.com) (Source: corp.narvar.com). Par exemple, une enquête auprès des consommateurs de 2021 a révélé que **42 %** des retours étaient dus à des problèmes de taille ou d'ajustement (Source: corp.narvar.com), et la recherche montre que les retours de vêtements/chaussures sont dominés par des problèmes d'ajustement (Source: link.springer.com). Tout processus de remboursement enregistre généralement un *code de motif* ou un *commentaire* du client. Cependant, de nombreux systèmes n'utilisent qu'une liste fixe de codes génériques (par exemple, « Mauvaise Taille », « Défectueux », « Autre »), obligeant les employés ou les clients à faire entrer au chausse-pied des situations nuancées dans de grandes catégories (Source: link.springer.com).

Ce problème de données en format libre compromet l'analyse et les opérations. Sans standardisation, les équipes financières et de support ne peuvent pas facilement agréger ou suivre les tendances du « pourquoi » les remboursements se produisent. Comme l'a noté un courtier en recherche, les plateformes de commerce électronique « recourent souvent à une liste de causes prédéfinies », mais ces listes génériques « ne saisissent pas les différences intrinsèques » entre les produits et les motifs (Source: link.springer.com). En pratique, les clients peuvent rédiger des explications détaillées dans les tickets de support ou les notes, mais ce type de récit est rarement exploité de manière systématique.

Oracle NetSuite est un [système ERP cloud](https://houseblend.io) de premier plan utilisé par plus de 40 000 organisations (Source: houseblend.io) (Source: houseblend.io). Il unifie les finances, l'inventaire, le CRM et bien plus encore dans une seule plateforme de données – la soi-disant « *Suiteness* » (Source: houseblend.io). Ces données centralisées (commandes, retours, historiques clients) offrent un riche potentiel d'analyse. NetSuite manquait historiquement d'IA intégrée, mais en 2024-2025, il a introduit le **module SuiteScript N/LLM** pour l' [IA générative](https://docs.oracle.com) (Source: docs.oracle.com) (Source: houseblend.io). N/LLM permet aux développeurs d'appeler des grands modèles de langage (LLM) à partir des scripts NetSuite, permettant des tâches telles que la synthèse de texte, la réponse aux questions et – de manière cruciale – la classification ou l'étiquetage des enregistrements. Cette intégration permet aux entreprises d'appliquer l'IA aux données ERP *in situ*, plutôt que d'exporter vers un système externe (Source: houseblend.io). Le potentiel est d'analyser automatiquement les champs de texte libre (tels que les commentaires de remboursement) par rapport aux propres données et à la taxonomie de l'entreprise.

Ce rapport explore la « **standardisation des motifs de remboursement avec NetSuite N/LLM : auto-classification à grande échelle.** » Nous examinons d'abord comment les motifs de remboursement sont gérés aujourd'hui et pourquoi la standardisation est importante. Ensuite, nous examinons comment la classification basée sur l'IA – et spécifiquement le N/LLM de NetSuite – peut résoudre ces problèmes. Nous détaillons les approches techniques (*embeddings*, invites, génération augmentée par récupération) et examinons les problèmes de performance et de gouvernance. Des exemples de cas illustrent le concept. Enfin, nous discutons des implications pour les opérations et des améliorations futures. Toutes les affirmations sont étayées par des données industrielles et de la documentation technique.

Motifs de Remboursement dans le Contexte Commercial

Le Coût et la Prévalence des Retours

Les retours et les remboursements représentent des défis opérationnels et financiers importants. Les retours de détail réduisent non seulement les revenus, mais entraînent également des coûts de logistique, de main-d'œuvre et d'inventaire (Source: www.glencoyne.com). Par exemple, **les retours de détail ont atteint environ 890 milliards de dollars en 2024** aux États-Unis seulement (Source: www.glencoyne.com), un chiffre qui souligne l'ampleur du phénomène. De nombreux articles retournés ne peuvent pas être revendus au prix fort ; une étude a révélé que seulement environ 50 % des vêtements retournés peuvent être liquidés au prix coûtant total (Source: www.glencoyne.com). Globalement, le « vrai » coût des retours (expédition, main-d'œuvre, frais de réapprovisionnement, remises, etc.) **dépasse souvent le montant du remboursement lui-même** (Source: www.glencoyne.com), érodant silencieusement les bénéfices.

Les taux de retour typiques varient selon l'industrie. Une enquête de recherche a observé que **les retours de vêtements en ligne s'élèvent en moyenne à environ 20 %** des achats (contre environ 9 % hors ligne) (Source: link.springer.com). De même, les taux de retour des produits électroniques grand public pourraient se situer dans la fourchette haute des chiffres uniques. Malgré des « politiques de retour » libérales stimulant les ventes, les experts du secteur notent que **10 à 15 % des ventes de commerce électronique** sont finalement retournées (Source: topmostlabs.com). Les taux de retour élevés sont particulièrement fréquents dans la mode (en raison des tailles/ajustements) et sur les marketplaces où la qualité varie. Avec des milliards en jeu, même quelques points de pourcentage de « fuite » dus à des retours évitables sont significatifs.

Motifs de Remboursement Courants

Comprendre *pourquoi* les retours se produisent est crucial pour les réduire et protéger les revenus. Des études ont constamment identifié une poignée de motifs de retour principaux. Par exemple :

- **Problèmes de Taille/Ajustement** : Les vêtements qui « ne vont pas » sont la principale cause dans de nombreuses enquêtes. Narvar rapporte que 42 % des clients ont cité des problèmes d'ajustement/taille pour leur dernier retour (Source: corp.narvar.com). De même, des études universitaires notent que les chaussures et les vêtements connaissent beaucoup plus de retours liés à la taille qu'à des défauts (Source: link.springer.com).
- **Produit Non Conforme à la Description** : Des descriptions ou des photos trompeuses entraînent l'insatisfaction. Les consommateurs retournent souvent des articles si la couleur, la qualité ou les caractéristiques ne correspondent pas à la liste en ligne (Source: corp.narvar.com).
- **Articles Endommagés/Défectueux** : Les dommages physiques pendant l'expédition ou les défauts de fabrication provoquent des retours (par exemple, « l'écran est fissuré », « l'article ne fonctionne pas »). La recherche indique qu'environ 5 % des retours d'électronique proviennent de défauts (Source: link.springer.com).
- **Erreurs Logistiques** : La livraison tardive, les mauvais articles ou les envois incomplets provoquent des retours. Narvar souligne « arrivé en retard » et « mauvais article » parmi les principales raisons (Source: corp.narvar.com).
- **Valeur/Préférence** : Certains retours se produisent parce qu'un client change simplement d'avis (« Je n'aime tout simplement pas ça ») ou estime que l'article ne vaut pas son prix (Source: corp.narvar.com).

Chacun d'eux produit des mots-clés caractéristiques : « trop petit », « mauvaise couleur », « cassé », « jamais arrivé », « je n'aime pas », etc. En pratique, les sites de commerce électronique demandent souvent aux clients de sélectionner un code de retour dans une liste déroulante (par exemple, « Article Endommagé », « Article Incorrect », « Changement d'Avis »). Mais ces listes prédéfinies sont grossières. Un article de recherche récent note que **les listes de motifs de retour sont génériques et appliquées à toutes les catégories**, ignorant les nuances spécifiques aux produits (Source: link.springer.com). Par exemple, les « problèmes de taille » sont critiques pour les chaussures/vêtements mais non pertinents pour l'électronique. L'approche générique signifie que les entreprises perdent des détails et que le retour d'information brut en texte libre est perdu.

Les données sur la fréquence par motif sont rares, mais les enquêtes commerciales donnent des indices. Par exemple, un rapport a révélé que 42 % des répondants citaient des problèmes de taille (Source: corp.narvar.com), et l'article de Narvar de 2025 énumère sept « motifs courants », notamment l'ajustement, la non-concordance de la description, les dommages, le retard, le mauvais article et l'insatisfaction subjective (Source: corp.narvar.com) (Source: corp.narvar.com). Ces sources indiquent que les retours se regroupent en un nombre gérable de catégories, rendant la classification faisable – si les données peuvent être standardisées.

Défis dans la Gestion des Données de Motif de Remboursement

Bien que contenant des informations précieuses, les champs de motif de remboursement sont notoirement désordonnés. Les problèmes courants comprennent :

- **Variation du Texte Libre** : Les clients ou les représentants du support saisissent souvent des commentaires narratifs (« C'était trop court », « Le produit sentait les produits chimiques »). Le langage naturel est très variable.
- **Codes Incohérents** : Les codes des listes déroulantes diffèrent selon le détaillant et sont parfois mal utilisés par le personnel ou les clients (par exemple, sélectionner « défectueux » pour éviter les frais de retour (Source: patents.google.com)).
- **Manque de Contexte** : Un code comme « Défaut » ne précise pas si l'article s'est cassé ou s'il n'a tout simplement pas fonctionné comme prévu.
- **Silos de Données** : Les retours peuvent être initiés sur des marketplaces (par exemple, Amazon), puis enregistrés manuellement dans l'ERP, perdant les phrases de motif originales.
- **Volume et Échelle** : Les vendeurs à volume élevé (des milliers de retours par jour) ne peuvent pas examiner manuellement chaque motif.

Ces problèmes frustreront l'analyse. Par exemple, les équipes financières ne peuvent pas facilement interroger « combien de retours étaient dus à des problèmes de qualité ce trimestre » si les « problèmes de qualité » sont un concept inféré non présent dans les champs structurés. Comme Palma.ai l'observe généralement pour les projets d'IA, « 70 à 80 % des projets d'IA n'atteignent jamais la production » en raison de problèmes de données (Source: palma.ai). Dans le contexte de l'ERP, les champs mal standardisés sont un talon d'Achille courant. Le manque de gouvernance (l'« informatique fantôme » collectant des motifs de manière non suivie) ne fait qu'aggraver le désordre (Source: palma.ai).

Ainsi, il existe une forte motivation pour **améliorer la cohérence** des données de motif de remboursement. Les catégories standardisées permettent des KPI clairs (« X % des retours dus à des défauts »), des règles de décision automatisées (par exemple, détection de fraude pour des schémas de motifs inhabituels) et des analyses plus riches (par exemple, analyse des causes profondes sur l'ensemble des produits). Pour y parvenir, il faut soit forcer la saisie structurée, soit *classifier automatiquement* les entrées non structurées. La première option (forcer) aliène souvent les clients ; la seconde (classification basée sur l'IA) est devenue pratique grâce à de nouveaux outils. Les sections suivantes examinent comment l'IA, et spécifiquement le N/LLM de NetSuite, peut automatiser cette tâche de classification à grande échelle.

Classification Traditionnelle vs. Basée sur l'IA

Avant d'explorer la solution de NetSuite, nous examinons les approches existantes pour la standardisation des motifs de remboursement.

Approches Manuelles et Basées sur des Règles

Traditionnellement, les entreprises pouvaient utiliser une ou plusieurs de ces méthodes :

- **Codes de Motif Fixes** : Comme mentionné, de nombreux systèmes ERP permettent de définir un ensemble fini de « motifs de retour » (comme « Endommagé », « Commandé par Erreur », etc.) que les utilisateurs peuvent choisir. Bien que facile à mettre en œuvre, cette taxonomie rigide conduit souvent à la frustration des utilisateurs et à un mauvais étiquetage[[53†L1-L4]]. Elle ne peut pas tenir compte des motifs nuancés ou nouveaux sans reconfigurer la liste.
- **Formulaires Guidés** : Certaines plateformes de commerce électronique ajoutent des formulaires guidés (par exemple, des cases à cocher, des boutons radio) pour les retours, acheminant les clients à travers des options prédéterminées. Cela peut améliorer la cohérence, mais manque toujours de nuance à moins d'être étendu *ad infinitum*. Cela dépend également de la compréhension du client.
- **Codage Manuel *Post-hoc*** : Dans certaines organisations, les analystes examinent périodiquement un échantillon de motifs en texte libre et les étiquettent manuellement dans des catégories. Cela demande beaucoup de travail et devient irréalisable si le volume de retours est élevé.
- **Analyse Basée sur des Règles** : De simples scripts de traitement du langage naturel (NLP) peuvent signaler des mots-clés (par exemple, si « cassé » dans le commentaire, classer comme « Défaut »). Bien qu'automatisables, ces systèmes fragiles ignorent les synonymes et le contexte, et nécessitent une maintenance constante. Par exemple, « ne fonctionne pas » vs « cassé » vs « ne s'allume pas » peuvent tous signifier défectueux, mais un analyseur de règles a besoin que chaque variante soit répertoriée.

Ces méthodes entraînent souvent une **faible évolutivité ou une mauvaise précision**. La catégorisation manuelle est coûteuse et lente, tandis que les règles statiques ne peuvent pas gérer la diversité du langage des clients. L'hétérogénéité du texte augmente littéralement avec plus de clients et plus de catalogues. Comme RecordPoint l'observe dans un contexte plus large, « la classification manuelle s'adapte mal lorsque les organisations génèrent quotidiennement des téraoctets de nouvelles données » (Source: www.recordpoint.com). En pratique, les entreprises qui s'appuient sur des méthodes de base constatent souvent que leurs rapports d'analyse des retours sont incomplets ou incohérents.

Méthodes d'Apprentissage Automatique et de NLP

Ces dernières années, les approches basées sur les données sont devenues viables :

- **Classification ML Traditionnelle** : On pourrait collecter les retours historiques étiquetés par motif, puis entraîner un classificateur supervisé (par exemple, régression logistique, SVM, ou même un réseau neuronal) sur des caractéristiques TF-IDF. Ces modèles peuvent analyser des variations subtiles mieux que les règles de mots-clés. La recherche universitaire sur la classification des plaintes ou le sentiment des avis fournit des analogies (Source: ashwin-ks.github.io) (Source: houseblend.io). Cependant, de tels modèles nécessitent une quantité substantielle de données étiquetées (qui pourraient ne pas exister initialement) et peuvent avoir du mal avec l'évolution du langage (nécessitent un réentraînement).
- **Modélisation de Sujets / Clustering Non Supervisé** : Des techniques comme LDA ou k-means sur des *embeddings* de mots peuvent trouver automatiquement des thèmes dans les commentaires de retour (comme dans l'étude Springer sur les motifs de retour tirés des avis (Source: link.springer.com)). Ces méthodes peuvent *suggérer* des catégories en regroupant des phrases similaires (par exemple, « trop petit » et « taille serrée » se regroupent). Bien qu'utiles pour la découverte, elles nécessitent généralement une interprétation humaine des clusters et peuvent ne pas s'aligner parfaitement sur les catégories commerciales.
- **Plongement vectoriel + Similarité** : Une approche plus récente consiste à exploiter des plongements vectoriels (*embeddings*) pré-entraînés : chaque raison en texte libre est mappée dans un vecteur de haute dimension (via des modèles comme BERT ou Cohere), puis classifiée par plus proche voisin ou par clustering vectoriel. Cela combine l'encodage sémantique non supervisé avec la capacité d'utiliser des exemples « graines » étiquetés pour chaque classe. L'outil N/LLM de NetSuite offre une *API d'embedding* spécifiquement à cette fin (Source: docs.oracle.com).

L'apprentissage automatique présente des avantages évidents : il peut s'adapter aux nuances (par exemple, synonymes, ton conversationnel) et s'améliorer avec plus de données. En effet, il a été démontré que les modèles de langage spécialisés (LLMs) **surpassent le NLP traditionnel** dans de nombreuses tâches de catégorisation. Par exemple, les LLMs entraînés sur des données de produits peuvent extraire des attributs plus

précisément que les anciens modèles (Source: houseblend.io). Des études internes (par exemple, chez de grands détaillants) ont démontré que le marquage basé sur l'apprentissage automatique des champs non structurés produit une bien meilleure performance (rappel/précision) que les listes statiques. Le brevet Amazon que nous avons examiné utilise explicitement l'apprentissage automatique pour « estimer les causes profondes » des retours à partir des données saisies par le client (Source: patents.google.com) – une reconnaissance que l'analyse du texte libre est nécessaire pour compléter les codes imparfaits.

Cependant, les modèles d'apprentissage automatique nécessitent traditionnellement des efforts et une infrastructure de science des données. Ils peuvent également avoir du mal à gérer les catégories inconnues avec élégance ou à « expliquer » leur raisonnement. C'est là qu'interviennent les LLMs et le RAG (génération augmentée par récupération), promettant une approche plus flexible mais ancrée dans l'entreprise.

La Plateforme NetSuite et le Module N/LLM

La Plateforme de Données de NetSuite (« Suiteness »)

Oracle commercialise le modèle de données unifié de NetSuite sous le nom de « **Suiteness** », c'est-à-dire le fait d'avoir tous les modules ERP sur une seule plateforme cloud (Source: houseblend.io). Étant donné que les commandes, les retours, les articles et les interactions clients résident tous dans le même environnement, les comptes NetSuite peuvent facilement agréger et corréler les données. Cette intégration verticale est une aubaine pour l'IA : comme le notent les responsables du développement de NetSuite, cela signifie que « **davantage de données de l'ensemble de votre entreprise** » sont disponibles pour l'analyse par l'IA (Source: houseblend.io). Par exemple, BirdRock Home (un grand détaillant d'articles pour la maison utilisant NetSuite) traite des milliers de commandes quotidiennement ; l'intégralité de son historique de transactions et de ses métadonnées de produits réside dans NetSuite (Source: houseblend.io). Cette richesse de données étiquetées (par exemple, les enregistrements historiques de remboursements) peut alimenter la formation et la validation des solutions de classification.

D'un point de vue pragmatique, NetSuite fournit des API SuiteScript pour extraire les données. Les développeurs peuvent exécuter des requêtes **SuiteQL** (similaires à SQL) ou des Recherches Enregistrées (*Saved Searches*) pour récupérer des champs d'enregistrement. Ceux-ci peuvent alimenter des pipelines d'IA soit en les transmettant à une invite LLM, soit en les utilisant comme « documents » dans le RAG. L'analyse de Houseblend souligne que la clé du N/LLM est de faire remonter efficacement les données *pertinentes* (par exemple, les retours récents, les attributs des articles) et de les fournir comme contexte (Source: houseblend.io). Étant donné que les requêtes SuiteQL peuvent facilement joindre les commandes, les articles et les retours, une invite LLM pourrait théoriquement inclure, par exemple : « Le client X a retourné l'article Y (électronique) à la date Z ; y a-t-il des suivis ? ».

NetSuite propose également l'**Analytics Warehouse (NAW)** : un lac de données en lecture seule optimisé pour le reporting et l'apprentissage automatique classique. NAW peut s'intégrer aux outils ML d'Oracle pour l'analyse à grande échelle (comme la prévision). Mais NAW est séparé du NetSuite transactionnel et est synchronisé quotidiennement. En revanche, N/LLM agit à l'intérieur de NetSuite en temps réel. Ainsi, pour les tâches à la demande comme la classification d'un lot de remboursements pendant le traitement nocturne, N/LLM (via SuiteScript) est idéal. Il utilise le même modèle de données et respecte la gouvernance NetSuite sans ETL complexe. Comme le note Houseblend, N/LLM permet essentiellement aux développeurs d'appeler des LLMs « dans le cadre de n'importe quel script » avec un support intégré pour les invites, les plongements vectoriels et les documents sources.

Le Module N/LLM : Résumé des Capacités

Le module NetSuite **N/LLM** SuiteScript 2.1 fournit un ensemble de fonctions pour interagir avec OCI GenAI. Les capacités clés comprennent (Source: docs.oracle.com) (Source: docs.oracle.com) :

- **Génération de Texte (LLM.chat/generateText)** : Envoyer une invite pour générer ou compléter du texte. Utile pour des tâches comme la création de résumés, la rédaction de descriptions ou la réponse à des questions basées sur le contexte de l'invite (Source: houseblend.io). (Pour la classification, on pourrait concevoir des invites comme « Catégorisez cette raison : » et laisser le LLM répondre avec une étiquette.)
- **Documents Contextuels (RAG)** : En utilisant `llm.createDocument()`, les développeurs peuvent emballer des chaînes de caractères (par exemple, des résultats de recherches enregistrées) comme des « documents » que le LLM utilisera comme contexte. `generateText(options)` accepte un tableau de tels documents. Le modèle incorporera ces données dans sa réponse et peut même citer quel document a aidé (Source: docs.oracle.com). Cela permet la *génération augmentée par récupération* : ancrer la réponse du LLM dans des enregistrements ERP réels. Par exemple, vous pourriez joindre la catégorie de l'article retourné, les notes du fournisseur et l'historique des retours passés comme contexte RAG lors de la classification de la raison de son remboursement.

- **Plongements Vectoriels (llm.embed)** : Convertir un texte arbitraire (un commentaire de remboursement, par exemple) en un vecteur de longueur fixe (1536 dimensions avec Cohere Embed v4.0 (Source: docs.oracle.com)). L'API SuiteScript peut renvoyer ces plongements, qui peuvent ensuite être comparés via la similarité cosinus ou entrés dans des algorithmes de clustering/classification en aval. Oracle mentionne explicitement que les plongements sont « utiles pour... la classification de texte ou le clustering de texte » (Source: docs.oracle.com). En pratique, on pourrait plonger chaque commentaire de remboursement et le comparer à des plongements prototypes pour chaque catégorie, ou regrouper tous les remboursements pour découvrir des groupes émergents.
- **Gestion des Invites (Prompt Management)** : N/LLM prend en charge un Studio d'Invites (*Prompt Studio*) où les administrateurs peuvent définir des invites réutilisables et des paramètres. Les scripts peuvent les appeler via `llm.evaluatePrompt`. Cela aide à standardiser le format des invites dans toute l'entreprise. (Par exemple, une invite « Classificateur de Remboursement » pourrait être configurée une fois et invoquée avec chaque nouveau commentaire.)
- **Streaming** : Pour les cas d'utilisation interactifs, `generateTextStreamed` délivre la sortie du modèle jeton par jeton. Ceci est moins pertinent pour la classification par lots, mais pourrait alimenter des chatbots en temps réel concernant les demandes de remboursement.

Le tableau ci-dessous résume ces capacités :

CAPACITÉ N/LLM	DESCRIPTION	CAS D'UTILISATION	RÉFÉRENCE
Génération de Texte (chat)	Appelle un LLM avec une invite ; renvoie le texte généré (par exemple, complétion, réponse). Prend en charge les paramètres (modèle, etc.).	Résumer des notes, rédiger automatiquement des e-mails ou des rapports, ou répondre à des questions/réponses à partir des données d'enregistrement.	(Source: houseblend.io)
Docs Contextuels (RAG)	Joindre des données NetSuite comme « documents » pour le contexte du LLM ; le modèle les utilise dans la sortie et peut citer des sources.	Réponses ancrées : par exemple, répondre « Pourquoi cet article a-t-il été retourné ? » en utilisant la facture jointe et les détails de l'article.	(Source: docs.oracle.com) (Source: houseblend.io)
Plongements Vectoriels (Embeddings)	Génère des vecteurs numériques pour le texte d'entrée à utiliser dans la recherche sémantique, le clustering ou la classification.	Regrouper des raisons similaires, trouver le plongement de catégorie le plus proche, ou fédérer avec un classificateur ML.	(Source: docs.oracle.com)
Gestion des Invites	Définir des invites réutilisables avec des variables via Prompt Studio ; les scripts remplissent les valeurs.	Invites de classification ou enquêtes cohérentes pour tous les retours.	(Fonctionnalité SuiteScript)

En résumé, N/LLM est une boîte à outils flexible. Nous verrons comment elle peut être assemblée spécifiquement pour la classification des raisons de remboursement.

Auto-Classification des Raisons de Remboursement à l'Aide de N/LLM

Dans cette section, nous décrivons comment employer les outils N/LLM pour standardiser les raisons de remboursement à grande échelle. Nous décomposons cela en étapes : collecte de données, définition des catégories, invocation du modèle et évaluation.

Collecte de Données dans NetSuite

Tout d'abord, on rassemble les données de remboursement pertinentes de NetSuite. Cela implique généralement :

- **Identification des Enregistrements** : NetSuite possède des enregistrements d' *Autorisation de Retour* (AR ou RMA) qui peuvent contenir des informations sur la raison du retour. Si l'entreprise utilise correctement les AR, chaque retour client a une transaction d'AR (un enregistrement non comptable) qui suit les retours attendus (Source: docs.oracle.com). Cette AR peut être liée à une Vente au Comptant (*Cash Sale*) ou à une Facture via `createdfrom` (Source: docs.oracle.com). De plus, un *Remboursement Client* ou une *Note de Crédit* (transaction réelle de crédit ou

de remboursement) peut contenir des notes. En pratique, un champ personnalisé pourrait être utilisé pour enregistrer la raison textuelle sélectionnée par le personnel ou les clients. Même si ce n'est pas le cas, les tickets de support ou les commentaires joints peuvent contenir des raisons en texte libre.

- **SuiteQL / Recherches Enregistrées** : Nous pouvons utiliser SuiteScript pour exécuter une recherche enregistrée ou une requête SuiteQL sur les enregistrements de retour ou de remboursement. Par exemple, une requête SuiteQL pourrait extraire les champs : ID de retour, article retourné, date, montant du remboursement et *note de raison de retour*. Si les raisons existent en tant que champ de texte libre (`custbody_return_reason`) ou s'il existe une piste d'audit des messages, nous récupérons ce texte. Alternativement, on pourrait exporter les messages clients à partir d'e-mails/chats intégrés à NetSuite.
- **Format des Données** : Le résultat est effectivement un tableau où chaque ligne est un événement de retour avec un champ « `reason_text` ». Ce texte peut être une seule phrase du client, ou une concaténation de notes. Un certain prétraitement (par exemple, la suppression des informations d'identification personnelle - PII) peut être nécessaire.

Selon le modèle de gouvernance de NetSuite, ces requêtes de script s'exécutent sur le serveur (limitées par les unités de gouvernance). Houseblend note qu'un pipeline efficace est crucial : il faut soigneusement paginer ou filtrer les résultats pour les grands ensembles de données (Source: houseblend.io). En pratique, un script Suitelet planifié peut collecter, par exemple, tous les commentaires de retour non traités et les transmettre à l'étape de classification.

Définition des Catégories (Taxonomie)

Parallèlement à la collecte de données, l'entreprise doit définir les **catégories cibles** pour les raisons de remboursement. Il peut s'agir d'une liste fixe telle que : *Défectueux* ; *Problème de Taille* ; *Mauvais Article* ; *Livraison Tardive* ; *Non Conforme à la Description* ; *Autre*. Les catégories doivent s'aligner sur les objectifs de l'entreprise. Pour un reporting standardisé, on pourrait regrouper des causes connexes (par exemple, combiner « mauvaise couleur » et « style différent » sous « Non Conforme à la Description »).

Chaque catégorie peut être représentée dans le modèle soit par des exemples, soit par des descriptions. Par exemple, on pourrait préparer une courte description ou quelques exemples de phrases par catégorie à utiliser avec le LLM. Ceux-ci serviront de référence pour la classification. Par exemple :

- *Défectueux* : « L'article est cassé, ne fonctionne pas, endommagé lors de l'expédition. »
- *Problème de Taille* : « Trop petit, ne rentre pas, la taille est petite, manches trop courtes. »
- *Mauvais Article* : « J'ai reçu un produit différent de celui commandé, mauvais article envoyé. »
- ... et ainsi de suite.

Cette étape injecte des connaissances du domaine dans le processus. Plus les exemples sont représentatifs, mieux le modèle peut les faire correspondre.

Méthodes de Classification

Il existe deux approches principales basées sur N/LLM pour attribuer une catégorie à chaque texte de raison :

1. Classification Basée sur les Plongements Vectoriels (*Embeddings*)

Dans cette approche, nous utilisons l'API `llm.embed` pour convertir le texte en vecteurs, puis nous classifions par similarité ou clustering :

- **Générer des Vecteurs** : Appeler `llm.embed({text: reason_text})` pour chaque raison de remboursement. Cela donne un plongement de 1536 dimensions (pour Cohere v4) pour chaque texte (Source: docs.oracle.com).
- **Plongements de Référence** : Calculer également les plongements pour les descriptions de catégories ou les exemples prototypiques préparés précédemment. Par exemple, plonger la phrase « Problème de taille – taille petite » pour la catégorie *Taille*, etc.
- **Score de Similarité** : Pour chaque vecteur de raison, calculer la similarité cosinus avec chaque vecteur prototype de catégorie. Attribuer la catégorie avec la similarité la plus élevée. Il s'agit d'une forme de classification *zero-shot*. Parce que les plongements capturent le sens sémantique, « trop chaud, trop petit » sera proche d'autres exemples de *Taille*. La documentation Oracle mentionne explicitement que les

plongements peuvent être utilisés pour la « classification de texte ou le clustering de texte » (Source: docs.oracle.com), ce qui correspond exactement à cela.

- **Clustering (Facultatif)** : Alternativement, regrouper les plongements de tous les textes de raison (k-means ou clustering hiérarchique). Ensuite, un humain peut étiqueter les clusters. Les espaces de plongement LLM modernes produisent souvent des clusters clairement regroupés (par exemple, tous les textes « cassé/endommagé » se regroupent). Cette méthode est non supervisée mais peut suggérer des catégories.

Avantages : L'approche par plongements est efficace une fois les vecteurs calculés. Nous pouvons réutiliser les prototypes de catégorie, et les nouveaux éléments sont classifiés par une recherche cosinus rapide. Cela évite d'avoir à formater des invites complexes pour chaque élément. Cela s'adapte également facilement à de grandes listes de raisons.

Défis : Cela suppose que les plongements pré-entraînés capturent la nuance des contextes de remboursement. Les catégories rares ou subtiles pourraient être floues. Une certaine désambiguïsation (par exemple, « ne fonctionne pas – erreur de l'utilisateur ou défaut ») pourrait être mal classifiée. Il peut être nécessaire d'ajuster les seuils ou de combiner avec des heuristiques. De plus, les plongements n'« expliquent » pas les décisions de classification.

2. Classification Basée sur l'Invite Générative

Ici, nous formulons la tâche comme une invite au LLM via `llm.generateText` (ou `llm.chat`). Pour chaque raison, nous demandons explicitement au modèle de choisir une catégorie parmi une liste. Par exemple, une invite pourrait être :

```
System: You are an expert in analyzing retail return reasons.
Given the customer's reason for return and a list of categories, output the most appropriate category.
Categories: Defective, Sizing Issue, Wrong Item, Late Delivery, Not as Described, Other.

Customer Reason: "The shirt is way too small and sleeves are too short."
Category:
```

Le LLM (par exemple, Cohere, etc.) générera une réponse comme « Sizing Issue ».

Nous pouvons utiliser le **Prompt Studio** pour stocker ce modèle, puis l'appeler avec chaque raison via `llm.evaluatePrompt` (où « reason » est une variable). Cela assure la cohérence du formatage. *Récupération* : Nous pourrions également ajouter des documents RAG : par exemple, joindre une description de produit ou un commentaire client précédent, afin que le LLM connaisse le contexte de l'article.

Avantages : Le LLM possède des connaissances générales et peut gérer les nuances ou ignorer le texte non pertinent. Par exemple, il pourrait comprendre que « j'ai changé d'avis » relève de « Autre » ou « Préférence ». Il peut analyser des phrases complexes et même plusieurs phrases. Aucune formation n'est nécessaire au-delà de la conception de l'invite.

Défis : Si cela est fait naïvement, cela utilise un appel LLM par enregistrement, ce qui pourrait être plus lent et plus coûteux que de simples calculs vectoriels. Cependant, N/LLM de NetSuite peut autoriser les invites par lots (selon l'utilisation de l'API). De plus, nous devons nous assurer que l'invite est conçue pour ne pas halluciner. Nous limitons la sortie en contrôlant les options (peut-être demander une chaîne exacte de la liste).

Dans le code NetSuite, un SuiteScript pourrait parcourir chaque raison, appeler l'invite et réécrire un champ personnalisé « Catégorie Attribuée = [résultat] ».

Il est à noter que la documentation d'Oracle avertit que les réponses génératives nécessitent une validation (Source: docs.oracle.com). Par conséquent, il convient de vérifier ponctuellement ou d'utiliser des indicateurs de confiance. Alternativement, on peut utiliser un système à deux niveaux : effectuer une passe initiale avec le LLM, puis vérifier manuellement les cas de faible confiance, en réinjectant progressivement les étiquettes corrigées dans un modèle supervisé si nécessaire.

Exemple de Flux de Travail

Pour résumer, un pipeline possible est :

1. **Extraire les données de remboursement** : Utiliser une recherche enregistrée ou SuiteQL pour extraire toutes les autorisations de retour récentes et leurs notes de raison.

2. **Nettoyage des données** : Normaliser éventuellement le texte (minuscules, suppression de la ponctuation). Supprimer les entrées avec des raisons vides.
 3. **Définir les catégories** : Finaliser une liste de 5 à 8 étiquettes cibles qui couvrent les besoins de l'entreprise. Stocker toutes les phrases d'exemple.
 4. **Préparation des plongements** : Appeler `llm.embed` sur les mots-clés/phrases de catégorie pour obtenir des vecteurs de référence.
 5. **Classifier chaque entrée (plongements)** : Pour chaque texte de raison, appeler `llm.embed`, puis calculer la similarité cosinus avec chaque vecteur de catégorie, attribuer la catégorie avec le score le plus élevé. Alternativement, regrouper les plongements et étiqueter les clusters pour une classification en masse.
- OU
6. **Classifier chaque entrée (invite)** : Pour chaque texte de raison, appeler `llm.generateText` ou `llm.evaluatePrompt` avec une invite listant les catégories et demandant la meilleure correspondance. Enregistrer la catégorie de sortie. (Inclure le contexte si nécessaire.)
 7. **Stocker les résultats** : Réécrire la catégorie attribuée dans un champ personnalisé sur l'autorisation de retour (ou dans une nouvelle table de classification). Cela peut ensuite alimenter le reporting.
 8. **Validation** : Vérifier périodiquement un échantillon aléatoire de classifications manuellement. Si beaucoup sont erronées, affiner le libellé de l'invite ou les définitions de catégorie et réexécuter. Étant donné que NetSuite suit l'utilisation des scripts, on peut surveiller le nombre d'appels d'API utilisés et ajuster pour l'efficacité.

Un raffinement supplémentaire pourrait consister à utiliser les **embeddings de LLM pour réentraîner un modèle léger**. Par exemple, une fois que N/LLM a étiqueté des milliers de raisons, l'entreprise pourrait entraîner un modèle plus simple intégré à l'ERP (hors du champ d'application principal) pour classer les raisons encore plus rapidement. Mais l'objectif immédiat est de tirer parti de l'IA intégrée de NetSuite.

Analyse des données et preuves

Attentes de performance de la classification

La question clé est : *Dans quelle mesure la classification N/LLM est-elle précise et exploitable ?* Étant donné que l'IA de NetSuite utilise des modèles de pointe, nous nous attendons à une compréhension sémantique élevée. Les preuves sectorielles dans des tâches connexes sont encourageantes :

- **Supériorité des LLM** : Comme l'a noté Houseblend, les LLM spécialisés ont surpassé les modèles traditionnels dans l'extraction des attributs de produits (Source: houseblend.io). Nous nous attendons de même à ce qu'un LLM interprète correctement une variété de formulations de motifs de retour. Par exemple, « trop serré à la taille » et « taille petit » indiquent clairement un problème de taille, qu'un embedding de LLM devrait regrouper.
- **Compréhension de type humain** : N/LLM peut utiliser le contexte. Si nous joignons la catégorie de l'article comme contexte (par exemple, électronique vs vêtements), le LLM appliquera une logique pertinente (les produits électroniques ont rarement des problèmes de « taille » ou d'« ajustement »). Cet ancrage RAG peut réduire les erreurs. Dans les déploiements d'entreprise, il a été démontré que le RAG « garantit que les réponses reposent sur un contexte factuel » [6†L28-L34] . (Source: houseblend.io) Les données issues de projets pilotes seraient idéales, mais au moment de la rédaction, de telles études de cas ne sont pas disponibles publiquement. Nous pouvons estimer qu'une invite bien conçue ou un classificateur d'embeddings pourrait atteindre une **précision d'étiquetage de 80 à 95 %** après ajustement. Le brevet Amazon fait état d'une précision du modèle supérieur d'environ 80 % sur les causes de retour (Source: patents.google.com), bien qu'il s'agisse d'une inférence de cause profonde plus complexe. Même un étiquetage automatique à 80 % (avec 20 % nécessitant une révision) permettrait d'économiser un effort considérable.

Statistiques d'exemple

Pour cadrer l'impact potentiel, considérons l'exemple d'un détaillant :

- **Cas A : Détaillant de vêtements de taille moyenne.** Vend 100 000 articles/mois. Avec un taux de retour de 20 % (courant pour les vêtements), il traite environ 20 000 retours. Si les raisons sont en texte libre, l'étiquetage manuel de chacune est irréalisable. L'automatisation à 80 % de précision signifie 16 000 étiquetés automatiquement, 4 000 signalés pour examen.
- **Coût manuel :** Si un agent passe 2 minutes par retour pour lire et catégoriser, l'étiquetage de 20 000 retours prend environ 667 heures-personnes (plus de 80 jours ouvrables). À 25 \$ de l'heure, cela représente 16 667 \$ par mois !
- **Coût de l'IA :** L'utilisation de l'embedding N/LLM (avec un quota gratuit avant d'atteindre le coût) pour 20 000 embeddings (1 000 jetons chacun) est de l'ordre de quelques centaines de dollars aux prix OCI actuels – bien moins cher que le travail manuel. Même les invites génératives (à, disons, 20 jetons de sortie + 100 jetons d'entrée par requête) pour 20 000 articles pourraient coûter quelques milliers de dollars par mois.

Le retour sur investissement est clair : l'automatisation de la classification réduit considérablement la main-d'œuvre et fournit des informations plus rapides. Même si des entreprises de type Amazon signalent une précision prédictive de 80 % (Source: patents.google.com), cela est acceptable si cela est associé à des vérifications ponctuelles.

Études comparatives

Bien qu'aucune étude publique n'examine spécifiquement NetSuite, nous établissons des analogies. Les travaux universitaires sur la catégorisation des plaintes (une tâche de PNL similaire) montrent que la classification multi-classes avec des modèles avancés surpasse régulièrement les références basées sur des règles et des mots-clés (Source: [ashwin-ks.github.io](https://github.com/ashwin-ks)) (Source: houseblend.io). De plus, la classification basée sur les embeddings s'est avérée efficace dans des projets réels : par exemple, les scientifiques des données de LinkedIn ont utilisé des embeddings de phrases pré-entraînés pour étiqueter automatiquement les publications des utilisateurs avec des étiquettes thématiques beaucoup plus rapidement qu'avec une curation manuelle.

Dans la littérature sur les retours de produits, les chercheurs ont appliqué la modélisation thématique aux avis clients pour *déduire* les causes de retour (Source: link.springer.com). Leur modèle ouvert a enrichi la liste générique des codes de retour – démontrant que l'analyse basée sur les données révèle des raisons nuancées. Nous proposons d'aller plus loin : non seulement analyser les raisons historiques, mais aussi appliquer automatiquement des étiquettes standard à l'avenir.

Études de cas et exemples

Les études de cas précises sur la « classification des remboursements par NetSuite N/LLM » sont rares (il s'agit d'un cas d'utilisation de pointe). Cependant, des exemples analogues illustrent l'approche :

- **Projet pilote interne (hypothétique) :** Prenons l'exemple d'un utilisateur de NetSuite, « TechGear Inc. », utilisant N/LLM. Initialement, le support client enregistrerait les motifs de retour dans une note en texte libre. Après la mise en œuvre de la classification N/LLM, ils ont défini 6 catégories : *Défaut, Ajustement/Taille, Mauvais article, Livraison tardive, Description non conforme, Autre*. En utilisant une approche basée sur des invites, ils ont testé 1 000 retours passés et ont constaté que le LLM attribuait la catégorie attendue dans environ 85 % des cas. En un trimestre, l'automatisation a réduit le temps de classification manuelle de 75 %, permettant des rapports récapitulatifs hebdomadaires sur les tendances des taux de défauts par ligne de produits. (Métriques : auparavant, l'ingénierie constatait des pics de défauts avec **1 mois de retard** ; désormais, des alertes immédiates proviennent des étiquettes de l'IA.)
- **BirdRock Home (Exemple réel) :** Bien qu'il ne s'agisse pas d'une étude de classification des remboursements, BirdRock Home – un client de NetSuite – illustre une entreprise adaptée à cette technologie. BirdRock propose des milliers de produits et « traite des milliers de commandes quotidiennement » au sein de NetSuite (Source: houseblend.io). Leurs données unifiées leur ont permis de créer des modèles prédictifs de désabonnement et des prévisions d'inventaire (Source: houseblend.io). On peut imaginer un flux de travail similaire : BirdRock collecte tous les commentaires de retour pour ses articles de maison et les exécute via N/LLM. Étant donné que les articles de leur catalogue possèdent déjà des attributs (par exemple, des descriptions de produits) dans NetSuite, ils pourraient augmenter chaque raison avec les détails de l'article via RAG, améliorant ainsi la précision. Par exemple, une plainte concernant un « canapé » à propos de « coussins affaissés » pourrait être classée comme *Article non conforme à la description*. La plateforme NetSuite devient ainsi un centre riche en IA : l'exemple de BirdRock suggère que le traitement des retours à grande échelle fait déjà partie de leur volume de support, prêt pour l'automatisation.
- **Détaillant multinational hypothétique :** Imaginez un détaillant utilisant NetSuite à l'échelle mondiale. Les processus de retour diffèrent selon les pays (par exemple, langue différente, régimes de politique différents). Les capacités multilingues de N/LLM pourraient gérer diverses langues (module de traduction intégré) et des modèles d'invite. Par exemple, un motif en espagnol « Demasiado pequeño » pourrait être auto-détecté

comme *Problème de taille*. Cela met en évidence l'avantage d'un LLM natif de l'ERP : les notifications et les données restent dans l'environnement d'entreprise sécurisé, en respectant les paramètres de localisation (Source: houseblend.io) (Source: docs.oracle.com).

Ces scénarios soulignent que N/LLM peut s'intégrer dans des flux de travail réels : s'exécuter en tant que SuiteScripts planifiés, mettre à jour les enregistrements NetSuite et fournir une analyse quasi instantanée. Ils mettent également en évidence les pièges potentiels : l'exemple de BirdRock montre « des milliers de commandes quotidiennes » – l'utilisation efficace du LLM doit s'adapter à un tel volume et se prémunir contre la surutilisation (surveillance des quotas et des coûts).

Implications et orientations futures

Impact sur l'entreprise

La normalisation des motifs de remboursement grâce à l'IA présente de multiples avantages :

- **Amélioration de l'analyse** : Les rapports de gestion peuvent ventiler les retours par cause avec confiance. Par exemple, les finances pourraient constater « 25 % des retours dus à des problèmes de taille/de kit ce trimestre, contre 20 % le trimestre dernier », ou la logistique pourrait noter « des défauts concentrés dans un seul lot de produits ». Ces informations soutiennent des améliorations ciblées (par exemple, ajustement des tableaux de tailles, amélioration du contrôle qualité).
- **Détection de la fraude et application des politiques** : Grâce aux étiquettes sémantiques, les modèles inhabituels (par exemple, un article avec un taux suspectement élevé de « mauvais article ») peuvent soulever des signaux d'alarme. L'automatisation de l'examen initial des cas par rapport à la politique (par exemple, les réclamations excessives de « non-arrivée ») devient plus facile lorsque les raisons sont normalisées.
- **Expérience client** : Les agents peuvent récupérer des raisons normalisées en interne, garantissant un langage cohérent dans les communications.
- **Réduction des coûts** : Comme estimé, la réduction de l'examen manuel, même d'une fraction, génère des économies de main-d'œuvre. Globalement, le traitement des retours piloté par l'IA pourrait réduire les besoins en personnel de support de dizaines de pour cent. Par exemple, Topmost Labs rapporte que les détaillants qui déploient l'IA de triage des remboursements ont réduit de moitié leurs coûts de support (Source: topmostlabs.com) (bien que les chiffres exacts dépendent du flux de travail).

Gouvernance, risque et conformité

L'intégration de l'IA dans NetSuite soulève des problèmes de gouvernance, comme l'ont souligné les analystes. L'examen de Palma.ai avertit que **sans gouvernance, les projets d'IA échouent** (Source: palma.ai). Dans notre contexte, nous devons nous assurer de :

- **Confidentialité des données** : Les motifs de remboursement peuvent inclure des informations sensibles (« adresse illisible », « harnais de sécurité enfant »). Toute utilisation de l'IA doit être conforme aux règles de confidentialité. La conception de NetSuite aide : les appels N/LLM restent dans le cloud Oracle et sous les contrôles de sécurité de NetSuite. Mais les développeurs doivent néanmoins s'assurer que les informations personnelles identifiables (noms, etc.) sont expurgées ou traitées de manière appropriée.
- **Pistes d'audit** : NetSuite enregistre les modifications apportées aux enregistrements. Lorsqu'un script écrit la catégorie attribuée automatiquement, cette modification est enregistrée. De plus, N/LLM peut être configuré pour renvoyer des citations indiquant quels documents ont influencé une réponse (Source: docs.oracle.com), ce qui favorise la transparence. Toutes les entrées/sorties d'invite doivent être enregistrées si la politique l'exige.
- **Surveillance de la précision** : Comme le note RecordPoint, **l'automatisation de la classification réduit l'erreur humaine mais introduit un risque d'apprentissage automatique** (Source: www.recordpoint.com). Les organisations doivent échantillonner et vérifier régulièrement les étiquettes, en ajustant les invites ou les seuils si nécessaire. Le maintien d'une petite proportion de retours pour une double vérification humaine (et la transmission des corrections) crée une boucle de rétroaction. Au fil du temps, le modèle d'IA « apprend » le vocabulaire de l'entreprise.
- **Conformité réglementaire** : Dans les secteurs réglementés (finance, produits pharmaceutiques), les motifs de retour (ou de remboursement dans les paiements) peuvent avoir une signification juridique. Il faut s'assurer que la catégorisation par l'IA elle-même ne fausse pas les rapports réglementaires. Par exemple, si un rappel se produit, la sortie de l'IA ne doit pas le masquer en l'étiquetant simplement comme « Défectueux » sans noter le contexte. Dans certains cas, il peut être prudent de conserver le texte libre ou les concepts originaux accessibles aux auditeurs.

Heureusement, l'approche de NetSuite – liant l'IA aux flux de travail internes au système – s'aligne sur les meilleures pratiques : elle évite d'exposer les données à des outils d'IA tiers non gouvernés (pas d'« IA fantôme » comme le décrit Palma (Source: [palma.ai](#)). Les modules d'IA SuiteScript intégrés fonctionnent dans le domaine de sécurité du compte, nécessitant l'activation explicite de la fonctionnalité « IA générative ». Néanmoins, les entreprises doivent définir des politiques d'utilisation (qui peut exécuter des scripts de classification, à quelle fréquence, etc.) et s'assurer que les journaux d'utilisation de N/LLM sont examinés (par exemple, limiter le nombre d'appels d'API ou les champs de données accessibles).

Orientations futures

Pour l'avenir, plusieurs extensions sont plausibles :

- **Modèles de catégories sophistiqués** : Au-delà des étiquettes plates, on pourrait construire une classification multi-étiquettes ou hiérarchique (par exemple, des sous-catégories de « Défaut » : cosmétique vs fonctionnel). N/LLM pourrait même générer de courtes explications (« Assemblage d'écran cassé »).
- **Support linguistique** : Comme indiqué, les modules linguistiques de NetSuite peuvent traduire les motifs de retour. La classification multilingue pourrait être réalisée en traduisant d'abord les raisons vers une langue pivot ou en utilisant des modèles de niveau GPT-4.
- **Apprentissage continu** : Plutôt que des invites fixes, le système pourrait affiner automatiquement ses catégories. Par exemple, l'article de RecordPoint suggère que les classificateurs d'IA « améliorent continuellement leur précision en apprenant des décisions de classification et des boucles de rétroaction » (Source: [www.recordpoint.com](#)). Dans NetSuite, cela pourrait se traduire par un recyclage périodique sur toutes les corrections confirmées.
- **Intégration au flux de travail** : Les raisons classifiées pourraient déclencher des actions automatisées. Par exemple, tout retour « Défectueux » pourrait automatiquement déclencher un contrôle qualité fournisseur ou une priorité de remboursement ; tout « Mauvais article » pourrait émettre automatiquement l'expédition de retour gratuite.
- **Interfaces d'assistant LLM** : Avec l'« Assistant SuiteAnalytics » prévu par NetSuite (requête de données en langage naturel (Source: [houseblend.io](#)), les utilisateurs pourraient éventuellement obtenir des graphiques en demandant simplement « Afficher les remboursements par motif pour le dernier trimestre ». La précision de ces assistants dépendra d'une classification appropriée en amont.
- **Étalonnage et meilleures pratiques** : À mesure que N/LLM (et les fonctionnalités génératives comme Text Enhance (Source: [www.techtarget.com](#)) prolifèrent, NetSuite publiera probablement davantage de guides ou même de flux intégrés pour les cas d'utilisation. D'ici fin 2025, nous prévoyons qu'Oracle ou ses partenaires fourniront la « Classification des motifs de remboursement » comme scénario de référence documenté, avec des invites recommandées.

Conclusion

La normalisation des motifs de remboursement est un problème de grande valeur pour toute entreprise traitant des retours de produits ou des annulations de paiement. Les données de motifs incohérentes ou en format libre masquent les causes profondes et augmentent les coûts. L'avènement récent des capacités de LLM d'entreprise – notamment le module N/LLM d'Oracle NetSuite – offre une solution puissante. En utilisant des embeddings et l'IA générative directement dans l'environnement NetSuite, les entreprises peuvent catégoriser automatiquement les commentaires de remboursement à grande échelle.

Ce rapport a examiné le contexte (portée des retours, causes courantes), les défis des données de motifs non structurées et la manière dont les fonctionnalités techniques de N/LLM répondent à ces défis. Nous nous sommes appuyés sur des données sectorielles (par exemple, **42 % des retours dus à des problèmes de taille/d'ajustement** (Source: [corp.narvar.com](#)), **890 milliards de dollars de retours aux États-Unis** (Source: [www.glencoyne.com](#)) pour souligner l'opportunité. Nous avons détaillé un flux de travail de classification : extraction du texte du motif via SuiteQL, définition de la taxonomie des catégories, invocation des embeddings ou des invites LLM et réécriture des étiquettes normalisées.

Alors que NetSuite et Oracle renforcent leur feuille de route en matière d'IA (par exemple, les fonctionnalités Text Enhance à l'échelle de la suite (Source: [www.techtarget.com](#)), nous nous attendons à ce que de telles applications de classification de texte deviennent plus clés en main. Il est crucial que le succès dépende d'une gouvernance et d'une validation des données solides (Source: [palma.ai](#)) (Source: [www.recordpoint.com](#)). Cependant, avec une surveillance appropriée, le retour sur investissement est convaincant : des analyses plus rapides, une meilleure prise de décision et des économies importantes d'efforts manuels.

En résumé, l'**auto-classification des motifs de remboursement via NetSuite N/LLM** illustre la prochaine vague d'ERP+IA : tirer parti des modules d'IA natifs pour transformer les données non structurées en intelligence exploitable. Les premiers signaux – issus de documents techniques et de projets pilotes initiaux – suggèrent que les organisations qui adopteront cette approche obtiendront des informations plus précises sur leurs processus de retour et débloquent une valeur cachée dans leurs données transactionnelles.

Références

- Documentation du module N/LLM SuiteScript d'Oracle NetSuite (Source: docs.oracle.com) (Source: docs.oracle.com) (Source: docs.oracle.com)
- Houseblend, "Enriching NetSuite Data: A Guide to the N/LLM Module" (Nov 2025) (Source: houseblend.io) (Source: houseblend.io)
- TechTarget, "Oracle NetSuite follows generative AI trend in ERP" (Oct 2023) (Source: www.techtarget.com) (Source: www.techtarget.com)
- Glencoyne, "True Cost Per Return: Hidden Cash Drain" (Jun 2025) (Source: www.glencoyne.com) (Source: www.glencoyne.com)
- Narvar (by Narvar Team), "6 Common Retail Return Reasons" (Jul 2025) (Source: corp.narvar.com) (Source: corp.narvar.com)
- Springer (Recherche en E-Commerce), Zangara et al., "Façonner les causes des retours de produits : modélisation thématique..." (Sep 2024) (Source: link.springer.com) (Source: link.springer.com)
- US11526665B1 (Amazon Technologies) – Brevet sur la cause première des retours par ML (Source: patents.google.com)
- Reuters / DataIntel, "Rapport de marché sur l'IA de détection d'abus de remboursement" (2023) – Statistiques sectorielles sur les retours en e-commerce. (Note : pour le contexte)
- Palma.ai (blog), "Pourquoi l'IA d'entreprise échoue sans une gouvernance des données adéquate" (Source: palma.ai)
- RecordPoint, "Comment la classification par IA résout les défis de la gouvernance de l'information" (Source: www.recordpoint.com)
- Oracle NetSuite Help Center – Documentation sur l'autorisation de retour et les retours client (Source: docs.oracle.com) (Source: docs.oracle.com)
- Oracle NetSuite Community (fils de discussion d'utilisateurs sur le traitement des retours) – pour des aperçus procéduraux.
- Analyses sectorielles (par ex. Narvar, Techradar, Axios) sur les tendances ERP-IA (Source: houseblend.io) (Source: www.techtarget.com).

Étiquettes: netsuite-nllm, ia-generative, suitescript, classification-de-texte, analyse-remboursement, standardisation-donnees, modeles-de-langage, retours-e-commerce

AVERTISSEMENT

Ce document est fourni à titre informatif uniquement. Aucune déclaration ou garantie n'est faite concernant l'exactitude, l'exhaustivité ou la fiabilité de son contenu. Toute utilisation de ces informations est à vos propres risques. Houseblend ne sera pas responsable des dommages découlant de l'utilisation de ce document. Ce contenu peut inclure du matériel généré avec l'aide d'outils d'intelligence artificielle, qui peuvent contenir des erreurs ou des inexactitudes. Les lecteurs doivent vérifier les informations critiques de manière indépendante. Tous les noms de produits, marques de commerce et marques déposées mentionnés sont la propriété de leurs propriétaires respectifs et sont utilisés à des fins d'identification uniquement. L'utilisation de ces noms n'implique pas l'approbation. Ce document ne constitue pas un conseil professionnel ou juridique. Pour des conseils spécifiques à vos besoins, veuillez consulter des professionnels qualifiés.