Terminology-Aware MT using Hybrid Rule-Based Methods and LLMs

Project Owner: David Wallgren | Course responsible: Staffan Larsson | Academic Supervisor: Arianna Masciolini | Student: Marco Leali

University of Gothenburg | Department of Philosophy, Linguistics and Theory of Science |
Master in Language Technology | LT2311 Language Technology Project

david.wallgren@volvo.com, staffan.larsson@ling.gu.se, arianna.masciolini@gu.se,
guslealma@student.gu.se

January 2026

Abstract

MT systems have advanced considerably in the past decade. However, their ability to handle domain-specific terminology remains limited, particularly in specialized technical domains. This paper presents a hybrid system for enhancing Machine Translation (MT) quality by integrating specific terminology with multi-stage post-processing. The system combines multiple NLP techniques, rule-based linguistic preprocessing, glossary enforcement, neural grammar correction and LLM-based fluency enhancement in a modular pipeline to produce high-quality English-to-Italian translations while preserving technical terminology consistency. Implemented in Python, the architecture leverages multiple open-source components including spaCy for linguistic analysis, Machine Translators, Hugging Face transformers (OmniGEC model) for Italian grammar correction, Ollama-based LLM (llama3.1 model) for contextual refinement and NLTK for evaluation. Evaluation using the metrics BLEU, ROUGE, BERT Score and TREU (Terminology Recall Evaluation Understudy) demonstrates some improvements in translation quality, especially regarding the use of terminology, the main scope of the project, but still the raw MT overall outperforms all the other approaches.

1. Introduction

The motivation of the project has its origin from the company Volvo Group, which was looking for a system able to automatically translate text using the company's terminology. For this purpose, the first step has been the identification of the terms that needed to be translated using the terms of the terminology database. The database's structure is composed by several metadata of terms in different languages regarding especially the automotive domain. In this project, the focus has been on the English and Italian language (the native speaker language of the author). To find the terms to be translated using the database, lemmatization and POS tagging have been used. Secondly, we explored different open sources Machine Translation (MT) systems, such as DeepL, LibreTranslate and Google Translate. The goal of the paper was not to find the best MT, but how to use it combing it with other approaches able to use the glossary terms. Even if MT systems have improved considerably in the last years, their ability to handle domain-specific terminology remains limited, particularly in specialized technical domains. Incorrect or inconsistent translations of technical terms can lead to significant misunderstandings and safety concerns in automotive engineering documentation. To address this limitation, this project implements and evaluates a four-stage hybrid translation system integrating terminology injection, rule-based linguistic preprocessing, neural grammar correction and LLM-based post-editing. The approach builds upon previous work on terminology-aware NMT [1][2][3][4][5] and demonstrates the significant potential of combining neural and LLM methods for producing accurate domain-specific translations. Throughout this paper, some of the questions that we will try to give an answer are: does glossary integration improve translation quality or just the use of MT is enough? How much does OmniGEC grammar correction help? What is the LLM enhancement contribution? Is the combined solution of the two approaches the best?

2. Related Work

Dougal and Lonsdale [1] showed that it is possible to improve translation quality by inserting the correct technical terms while the NMT model is generating the translation (decoding time). Michon et al. [2] use a similar idea, but rely on placeholders and constrained decoding to make sure the glossary terms appear in the final output. Chen [3], instead, uses LLMs and shows that it is possible to influence terminology usage through prompting, without modifying the underlying decoding process. Semenov [4] analyzes the WMT 2023 Shared Task and highlights the need for clear benchmarks and consistent terminology handling. Lackner [5] reviews how different terminology formats (TBX, glossaries, structured metadata) affect in-context learning in LLMs, showing how NMT and LLM methods are starting to converge. The approach of this project builds on all these ideas by combining them into one workflow:

1. Like Dougal and Lonsdale (2020), terms are injected into the translation and TREU metric is used for the evaluation.
2. Like Michon et al. (2020), rules and constraints are used to place the terms correctly.
3. Like Chen (2023), two of the approaches rely on LLM prompting to refine terminology usage.
4. Inspired by Semenov (2023) and Lackner (2024), I used a POS filtering approach for the term injection.

Together, these components form a multi-stage pipeline that merges terminology injection, grammatical post-processing and LLM-based refinement in a single integrated system.

3. Dataset

The main data involved in this project are the following ones:

- The terminology dataset used for the glossary injection
- The translation memory of the Driver Guide from Volvo Trucks, used in the evaluation.

3.1 The Terminology Dataset

An Excel file (.xlsx), containing different columns, was extracted from the Volvo Terminology Database. The most relevant for this paper are the “en-GB” and “it” columns, since in this paper, we focused only on the English and Italian languages. In the code this file is uploaded as a dictionary with the following format:

```
lubricating gun -> {'it': 'pistola lubrificazione', 'sv': 'smörjspruta'}
```

where the keys are ‘it’ and ‘sv’ and the values ‘pistola lubrificazione’ and ‘smörjspruta’ respectively. The dictionary could be easily extended to take into consideration other languages or other parameters. In the first phase of the project, Swedish language was taken into consideration, but then, for the sake of simplicity, we decided to go on only with the Italian translation, the speaking native language of the author.

3.2 The translation memory

The translation memory is a database that stores sentences that have already been translated by humans (or validated by humans), so they can be reused in future translations. Volvo has provided us the translation memory (.tmx) of the driver guide EN-IT and it was used for the evaluation of our code. This file has been uploaded in the system returning a list of tuples in the following format:

```
translation_pairs = [
    ("English sentence 1", "Italian translation 1"),
    ("English sentence 2", "Italian translation 2"),
    ("English sentence 3", "Italian translation 3"),
                    # ... up to 9141 pairs in the file
            ]
```

The total of translations consists of 9141 pairs with an average English sentence length of 54 chars and average Italian sentence length of 65.5 chars.

4. Methodology

4.1 Translation System

The code can be split into two main components: the translation system and the evaluation framework (for the Evaluation see section §4.11). The implemented Python system follows a modular architecture organized into different processing phases, see Figure 1. Primarily, the TMX data and the terminology glossary are processed and the input text is pre-processed (including Saxon genitive handling). At this point, we can already produce the first baseline output using raw MT, which does not yet use any terminology injection beyond the initial glossary loading. Secondly, the system searches for glossary terms in each sentence and marks them. MT is then applied again, this time using a placeholder mechanism combined with Italian morphological adjustments to handle inflection and multi-word terms. After this intermediate translation, we arrive to the core of the project. Two separate post-processing paths are applied: OmniGEC [9], which corrects grammar errors (producing Output 2); LLM refinement, which improves fluency and terminology usage through prompting (Output 3). A combined approach is also tested: after OmniGEC correction, the LLM is applied to the corrected text, producing Output 4. Finally, all outputs are evaluated with multiple metrics to assess translation quality and terminology consistency. The code is available at https://github.com/lealimarco/Advanced_MT.

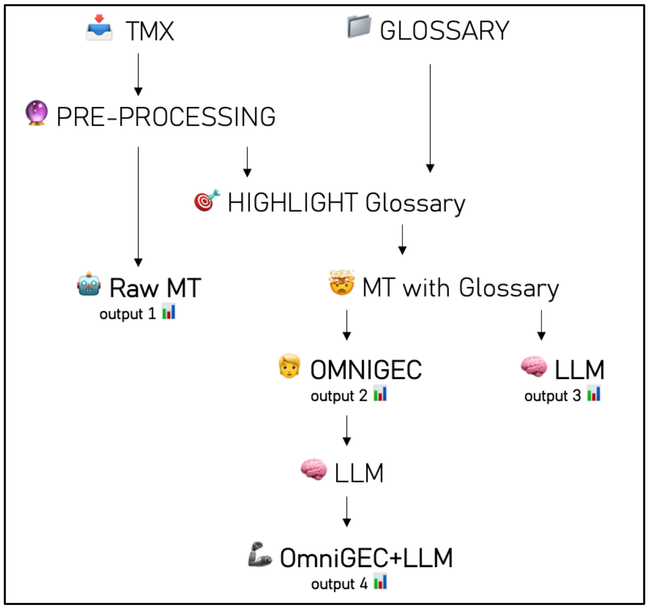


Figure 1: System Architecture: workflow from English source text through multiple enhancement stages to final Italian translation.

4.2 Translation Memory eXchange Loading

The TMX loading phase serves as the data ingestion component of the translation pipeline, responsible for processing and preparing bilingual translation memory data for system evaluation. This module handles the extraction of aligned English-Italian sentence pairs from standardized TMX files, which form the primary dataset for performance assessment.

4.3 Pre-Processing

The preprocessing stage employs regular expressions and spaCy dependency parsing to convert Saxon Genitive forms into prepositional phrases when they occur (e.g. "truck's service cover" becomes "service cover of the truck"), significantly improving syntactic compatibility with Italian structures. For the upcoming translations, excluding the pure MT translation given by output 1, a system of placeholders will be used. For this reason, the removal of any Saxon Genitive reference ('s) would be optimal for this approach.

4.4 Terminology Management

The glossary loader extracts triplets from the Volvo terminology dataset (.xlsx) containing English, Italian and Swedish terms, stored in a nested dictionary structure. The system includes comprehensive terminology validation and fallback mechanisms for missing translations.

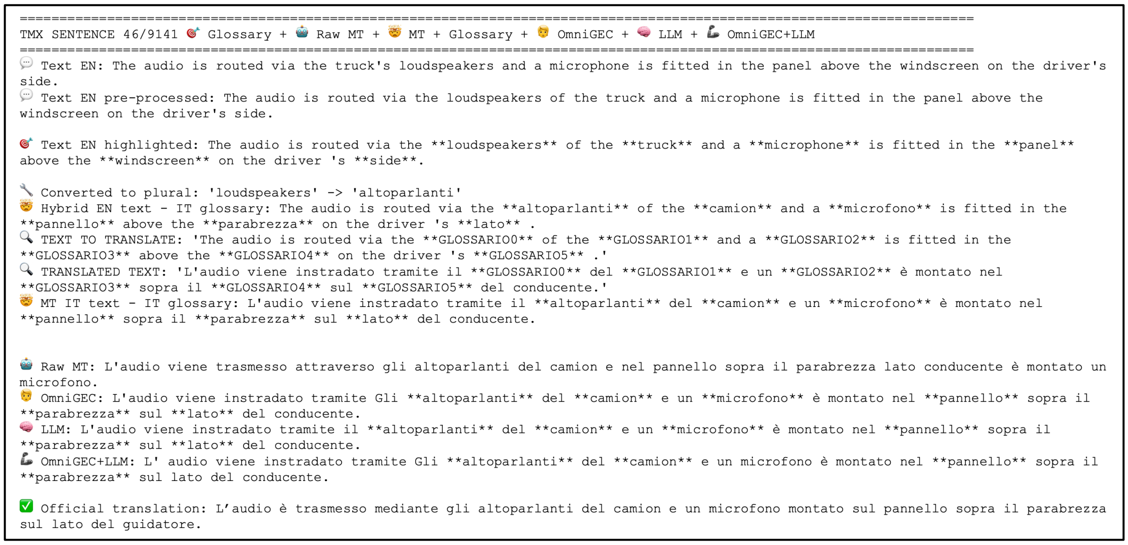
4.5 Terminology Highlighting

In this section spaCy lemmatization and part-of-speech (POS) tagging are used to identify single-word and multi-word (longest match first strategy) technical terms for substitution. The system implements longest-match-first strategy and handles verb/noun differentiation through the morphological metadata given from the lexicon. For the sake of simplicity, POS allowed have been only "NOUN", "PROPN", "ADJ", "VERB" and "AUX". English terms that have empty Italian translation are skipped and not highlighted.

4.6 Terminology Injection and Machine Translation (pre-phase for output 2-3-4)

In the pre-phase step for outputs 2-3-4, , glossary terms are replaced by terminology placeholders (e.g. "fuse box" becomes "GLOSSARIO0") with glossary-defined equivalents (e.g. "scatola fusibili"). This function performs terminology-aware translation through placeholders:

1. Input processing: it receives as input the output coming from the Terminology Highlighting step, where the terminology terms are between the asterisks "**".
2. Term extraction and translation: for each highlighted word, we check if they are in the plural form or not, using POS tagging and, as fallback, we check if the end of the English word is with "s", but not with "ss". Then, the main function lemmatizes the highlighted word and find the correspondent Italian translation. If the English word was plural, basic Italian rules for plural inflection are applied on all the highlighted words. Glossary terms are replaced (see Figure 2).
3. Possessive syntax correction: it analyses the text and goes to apply corrections in case it detects consecutive **term1** **term2** patterns with possessive constructions (e.g. "of the", etc), applying possessive syntax.
4. Placeholder protection: at this point, it replaces the highlighted Italian terms with a system of unique placeholders (GLOSSARIO0, GLOSSARIO1, …, GLOSSARIOn).
5. Machine translation: MT translates the text with the placeholders' system. This is crucial for preserving specialized terminology.
6. Term restoration: finally, glossary terms are restored instead of the placeholders. The final outcome is the sentence in full Italian language, but likely with some grammatical errors due to the injection and placeholder's process.



```
=====================================================================================================
TMX SENTENCE 46/9141 🎯 Glossary + 🤖 Raw MT + 🤯 MT + Glossary + 👦 OmniGEC + 🧠 LLM + 🦾 OmniGEC+LLM
=====================================================================================================
💬 Text EN: The audio is routed via the truck's loudspeakers and a microphone is fitted in the panel above the windscreen on the driver's
side.
💬 Text EN pre-processed: The audio is routed via the loudspeakers of the truck and a microphone is fitted in the panel above the
windscreen on the driver's side.

🎯 Text EN highlighted: The audio is routed via the **loudspeakers** of the **truck** and a **microphone** is fitted in the **panel**
above the **windscreen** on the driver 's **side**.

🔧 Converted to plural: 'loudspeakers' -> 'altoparlanti'
🤯 Hybrid EN text - IT glossary: The audio is routed via the **altoparlanti** of the **camion** and a **microfono** is fitted in the
**pannello** above the **parabrezza** on the driver 's **lato** .
🔍 TEXT TO TRANSLATE: 'The audio is routed via the **GLOSSARIO0** of the **GLOSSARIO1** and a **GLOSSARIO2** is fitted in the
**GLOSSARIO3** above the **GLOSSARIO4** on the driver 's **GLOSSARIO5** .'
🔍 TRANSLATED TEXT: 'L'audio viene instradato tramite il **GLOSSARIO0** del **GLOSSARIO1** e un **GLOSSARIO2** è montato nel
**GLOSSARIO3** sopra il **GLOSSARIO4** sul **GLOSSARIO5** del conducente.'
🤯 MT IT text - IT glossary: L'audio viene instradato tramite il **altoparlanti** del **camion** e un **microfono** è montato nel
**pannello** sopra il **parabrezza** sul **lato** del conducente.


🤖 Raw MT: L'audio viene trasmesso attraverso gli altoparlanti del camion e nel pannello sopra il parabrezza lato conducente è montato un
microfono.
👦 OmniGEC: L'audio viene instradato tramite Gli **altoparlanti** del **camion** e un **microfono** è montato nel **pannello** sopra il
**parabrezza** sul **lato** del conducente.
🧠 LLM: L'audio viene instradato tramite il **altoparlanti** del **camion** e un **microfono** è montato nel **pannello** sopra il
**parabrezza** sul **lato** del conducente.
🦾 OmniGEC+LLM: L' audio viene instradato tramite Gli **altoparlanti** del **camion** e un microfono è montato nel **pannello** sopra il
**parabrezza** sul lato del conducente.

✅ Official translation: L’audio è trasmesso mediante gli altoparlanti del camion e un microfono montato sul pannello sopra il parabrezza
sul lato del guidatore.
```

Figure 2: Example of handling plural form.

4.7 Italian Syntax Refining – LLM-based Grammar Correction (output 2)

The system integrates the OmniGEC-Minimal-8B model through Hugging Face transformers for specialized Italian grammar correction. This component handles complex syntactic structures while preserving marked terminology. To understand how OmniGEC works, we should make a step back and define what it is GEC. GEC is the AI task of automatically detecting and correcting grammatical errors in text, transforming sentences like "He go to school" to "He goes to school." It uses machine learning models trained on parallel datasets of incorrect/correct sentences. OmniGEC [9] introduces a silver-quality multilingual GEC dataset created through back-translation, enabling training of models across 10 languages (including Italian and English). The key innovation is using machine translation to generate artificial grammatical errors that mimic native speaker mistakes, then correcting them to create training pairs. This scalable approach addresses the scarcity of high-quality GEC data for many languages, particularly for technical domains where professional translations exist. OmniGEC-Minimal-8B is the model trained on this dataset: a large multilingual transformer fine-tuned for grammar correction.

4.8 LLM Enhancement Setup (output 3)

As alternative approach to the one described in Section §4.7, Ollama, a tool that let to run large language models locally, has been used with the model LLaMA 3.1. The scope was to refine fluency trying to preserve the glossary terms through a prompt. The enhancement includes contextual adaptation of terminology for number/gender while maintaining base term integrity. Different attempts have been made to improve the prompt in ordert to preserve the glossary. Nevertheless, glossary terms, in some occasions, are still changed.

4.9 Combined Approach: OmniGEC and LLM (output 4)

It is a combination of the two above described approaches: first omniGEC model is applied and later the LLM with the same prompt.

4.10 System Architecture Example

Figure 3 shows an overall example of how all the implemented system works.

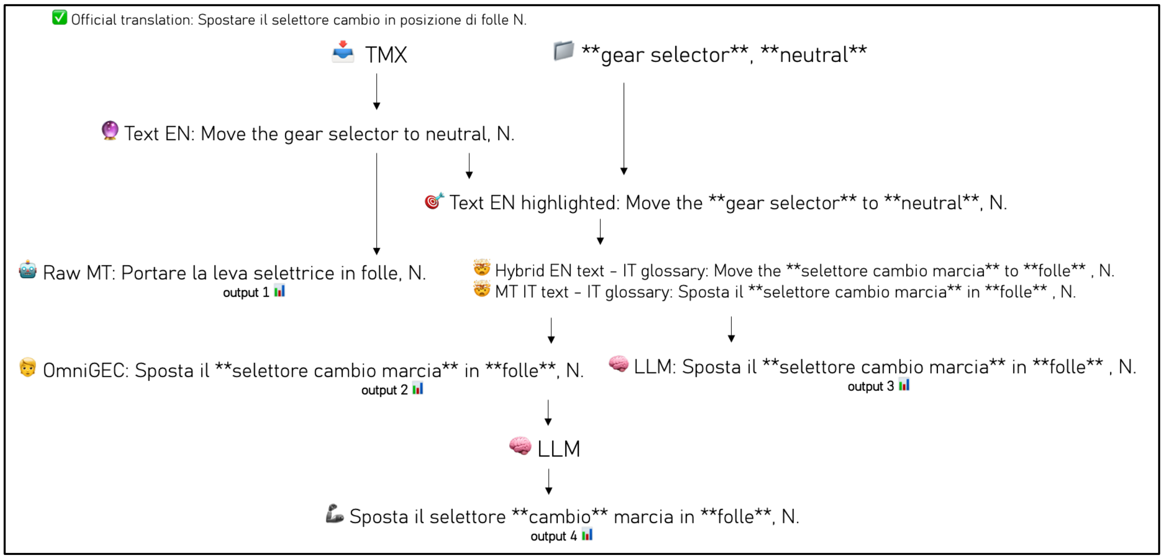

✅ Official translation: Spostare il selettore cambio in posizione di folle N.
TMX
gear selector, **neutral**
Text EN: Move the gear selector to neutral, N.
Text EN highlighted: Move the **gear selector** to **neutral**, N.
Raw MT: Portare la leva selettrice in folle, N.
output 1
Hybrid EN text – IT glossary: Move the **selettore cambio marcia** to **folle** , N.
MT IT text – IT glossary: Sposta il **selettore cambio marcia** in **folle** , N.
OmniGEC: Sposta il **selettore cambio marcia** in **folle**, N.
output 2
LLM: Sposta il **selettore cambio marcia** in **folle** , N.
output 3
LLM
Sposta il selettore **cambio** marcia in **folle**, N.
output 4

Figure 3: Processing example for the original English text: “Move the gear selector to neutral, N.”

Raw MT produces the first output by translating the pre-processed sentence directly, without consulting the terminology database. The remaining three approaches follow a different workflow. First, the system identifies the relevant terms contained in the terminology database. Then, these terms are injected into the translation process, resulting in Output 2 and Output 3. Finally, Output 4 is generated by applying an LLM-based enhancement to Output 2, producing the combined approach, where both terminology injection and LLM post-editing are integrated.

4.11 Evaluation Framework

Evaluation used 1,000 sentences (from the total available of 9,141 sentences) from Volvo's TMX translation memory (the Driver Guide from Volvo Trucks), with official Italian translations as references and with 3 runs for each sentence. Three runs have been used to test consistency of each approach, capturing the variability in stochastic components. The system compares 4 different approaches for each sentence:

1. Raw MT - Baseline Google Translate
2. OmniGEC - MT + Grammar Correction
3. LLM - MT + Ollama Enhancement
4. OmniGEC+LLM - Combined approach

Four different evaluation metrics different have been considered: BLEU [Papineni et al., 2002], ROUGE [Lin, 2004], BERTScore [Zhang et al., 2019] and the implemented metric TREU (Terminology Recall Evaluation Understudy) [Dougal et al., 2020]. While in literature standard metrics like BLEU, ROUGE and BERTScore are well-known and used for MT evaluation (see Table 1), they have some limitations for a task like the one of this paper, which is looking for a terminology accuracy. To address this critical gap, taking as reference the paper of Dougal et al., 2020, I implemented a metric inspired by metric TREU (Terminology Recall Evaluation Understudy). This metric specifically addresses terminology evaluation accuracy by combining orthographic overlap with terminology credit for correctly used glossary terms. Some key aspects of my implementation are:

1. Lemmatized matching which handles grammatical variations automatically
2. Separate terminology credit in order that glossary terms get extra weight
3. Credit-based scoring that combines with standard overlap
4. Detailed reporting of injected terms for analysis

Metric	Purpose	What it measures
BLEU	General translation quality	n-gram overlap with reference
ROUGE	Content coverage	Overlap of n-grams and longest sequences
BERTScore	Semantic similarity	Contextual embedding similarity
TREU	Overall translation + terminology quality	General translation quality + terminology usage credit

Table 1: Evaluation Metrics Overview

5. Results and Discussion

The evaluation across 3000 translation instances (1000 sentences for 3 runs) reveals clear performance patterns with raw machine translation (Raw MT) emerging as the dominant approach, achieving 73.2% selection rate as the best-performing method. In other words, across 3000 translation instances, 2197 times (73.2%) raw MT gave the highest overall average score between all the four approaches.

- Raw MT: 2197 times (73.2%)
- OmniGEC: 588 times (19.6%)
- LLM: 177 times (5.9%)
- Combined: OmniGEC + LLM: 38 times (1.3%)

Giving a look to the average score gained for each approach between all processed sentences and making an average between the four metrics, we get that the Raw MT leads with an overall average of 67.5 % (see Table 2).

APPROACH	BLEU	ROUGE-L	BERT	TREU F1	AVERAGE
🤖 Raw MT	44.5±29.9 (67.3%)	69.9±24.2 (34.6%)	77.6±17.6 (22.6%)	78.0±23.8 (30.6%)	67.5±21.6 (32.0%)
👨 OmniGEC	37.4±28.4 (75.9%)	64.4±25.3 (39.3%)	70.8±20.4 (28.8%)	74.7±25.7 (34.4%)	61.8±22.1 (35.7%)
🧠 LLM	35.9±28.2 (78.5%)	63.7±24.7 (38.8%)	71.2±19.2 (27.0%)	73.6±24.5 (33.3%)	61.1±21.0 (34.4%)
🦾 OmniGEC + LLM	37.2±28.0 (75.4%)	64.5±25.1 (38.9%)	70.3±19.8 (28.2%)	70.9±25.9 (36.5%)	60.7±21.6 (35.5%)

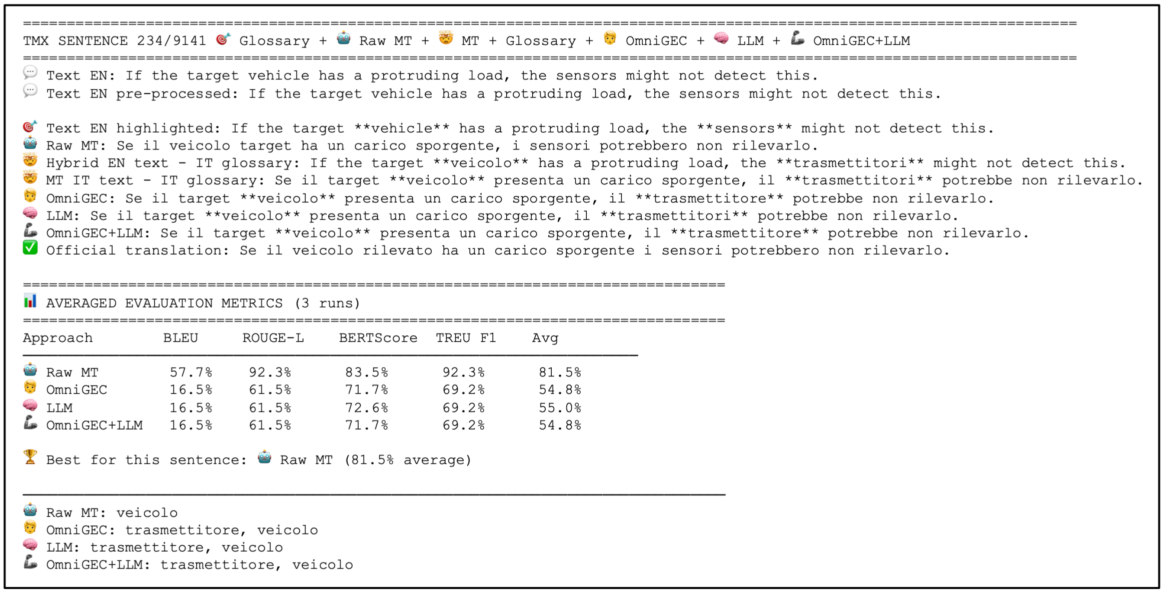
Table 2: Overall averages of scores for each approach across all the 3000 sentences.

For each average, Standard Deviation (±) and Coefficient of Variation (CV) in the brackets have been calculated. Standard Deviation measures how much individual scores vary from the mean while CV measures the relative variability ((Standard Deviation / Mean) x 100%). The lower they are the more consistent and stable the performance is. The raw MT approach demonstrates not only higher average performance, but also superior consistency with the lowest variation across different sentences.

Contrary to initial hypotheses, the unmodified Google Translate API consistently outperformed all enhancement pipelines. This suggests:

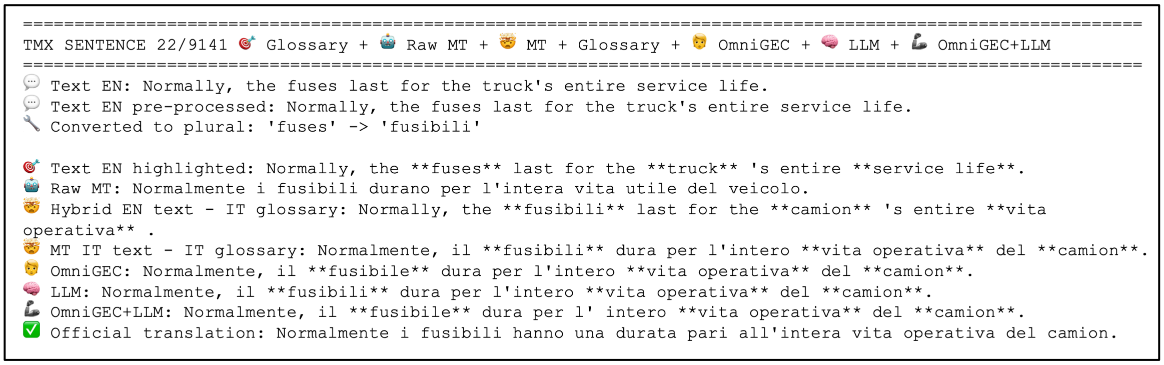
1. API Maturity: MT systems like Google Translate have undergone extensive optimization for general-domain translation
2. Pipeline complexity overhead: each additional processing stage introduces potential error propagation
3. Terminology interference: aggressive glossary injection may disrupt the MT system's internal coherence mechanisms

The results demonstrate that in technical translation scenarios, apparently, simpler often proves better. The raw Google Translate API's consistent outperformance suggests that current MT systems have reached a good level where extensive pre- and post- processing may degrade rather than improve quality. This is confirmed also if we move our focus just on the TREU metric, which for the scope of the project is considered the most relevant, instead of looking to an average of the metrics results. Raw MT is still the best approach with 78.0% ± 23.8 (CV: 30.6%) F1 score. On the other side, all the other three approaches are not so far from getting the same score and, as we discussed, the other metrics are not well-suited for the scope. Moreover, for the final evaluation, there is a very crucial point to be considered. After a human evaluation of outputs on some selected phrases, we noticed that it can also occur the case where the official Italian translation does not use the official terminology given by the glossary database (see Figure 5). This could affect the results considerably, especially when considering the TREU metric. So, given the used metric evaluation, MT result to be the best approach, but practically, after a manual inspection on some of the sentences, this does not seem to be always true.



```
================================================================================================================
TMX SENTENCE 234/9141 🎯 Glossary + 🤖 Raw MT + 🤯 MT + Glossary + 👨 OmniGEC + 🧠 LLM + 🦾 OmniGEC+LLM
================================================================================================================
💬 Text EN: If the target vehicle has a protruding load, the sensors might not detect this.
💬 Text EN pre-processed: If the target vehicle has a protruding load, the sensors might not detect this.

🎯 Text EN highlighted: If the target **vehicle** has a protruding load, the **sensors** might not detect this.
🤖 Raw MT: Se il veicolo target ha un carico sporgente, i sensori potrebbero non rilevarlo.
🤯 Hybrid EN text - IT glossary: If the target **veicolo** has a protruding load, the **trasmettitori** might not detect this.
🤯 MT IT text - IT glossary: Se il target **veicolo** presenta un carico sporgente, il **trasmettitori** potrebbe non rilevarlo.
👨 OmniGEC: Se il target **veicolo** presenta un carico sporgente, il **trasmettitore** potrebbe non rilevarlo.
🧠 LLM: Se il target **veicolo** presenta un carico sporgente, il **trasmettitori** potrebbe non rilevarlo.
🦾 OmniGEC+LLM: Se il target **veicolo** presenta un carico sporgente, il **trasmettitore** potrebbe non rilevarlo.
✅ Official translation: Se il veicolo rilevato ha un carico sporgente i sensori potrebbero non rilevarlo.

================================================================================
📊 AVERAGED EVALUATION METRICS (3 runs)
================================================================================
Approach         BLEU      ROUGE-L     BERTScore  TREU F1     Avg
______________________________________________________________________
🤖 Raw MT          57.7%     92.3%       83.5%       92.3%       81.5%
👨 OmniGEC         16.5%     61.5%       71.7%       69.2%       54.8%
🧠 LLM             16.5%     61.5%       72.6%       69.2%       55.0%
🦾 OmniGEC+LLM     16.5%     61.5%       71.7%       69.2%       54.8%

🏆 Best for this sentence: 🤖 Raw MT (81.5% average)

________________________________________________________________________________
🤖 Raw MT: veicolo
👨 OmniGEC: trasmettitore, veicolo
🧠 LLM: trasmettitore, veicolo
🦾 OmniGEC+LLM: trasmettitore, veicolo
```

Figure 5: Example of “not approved” usage of the glossary. The official Italian translation does not use the official terminology detected correctly by the three approaches OmniGEC, LLM and the combined approach.



```
==========================================================================================================
TMX SENTENCE 22/9141 🎯 Glossary + 🤖 Raw MT + 🤯 MT + Glossary + 👦 OmniGEC + 🧠 LLM + 🦾 OmniGEC+LLM
==========================================================================================================
💬 Text EN: Normally, the fuses last for the truck's entire service life.
💬 Text EN pre-processed: Normally, the fuses last for the truck's entire service life.
🔧 Converted to plural: 'fuses' -> 'fusibili'

🎯 Text EN highlighted: Normally, the **fuses** last for the **truck** 's entire **service life**.
🤖 Raw MT: Normalmente i fusibili durano per l'intera vita utile del veicolo.
🤯 Hybrid EN text - IT glossary: Normally, the **fusibili** last for the **camion** 's entire **vita
operativa** .
🤯 MT IT text - IT glossary: Normalmente, il **fusibili** dura per l'intero **vita operativa** del **camion**.
👦 OmniGEC: Normalmente, il **fusibile** dura per l'intero **vita operativa** del **camion**.
🧠 LLM: Normalmente, il **fusibili** dura per l'intero **vita operativa** del **camion**.
🦾 OmniGEC+LLM: Normalmente, il **fusibile** dura per l' intero **vita operativa** del **camion**.
✅ Official translation: Normalmente i fusibili hanno una durata pari all'intera vita operativa del camion.
```

Figure 6: Problems with number agreement regarding the injected term.

We can conclude that the term injection is one of the most critical components of the system. Terms must be injected with the correct morphological form, especially with respect to inflection, number and multi-word terms often require additional adjustments. In some cases, certain terms should not be injected at all (e.g. the English term "sensor" is usually translated officially by "sensore" whereas in the database is identified as "trasmettitore"). To address these issues, several functions were implemented within the code to correct or adjust injected terms. However, we observed that conflicts still arise (see Figure 6). For example, even when the injected term is correctly converted to the plural, OmniGEC tends to revert it to the singular form to match the verb, likely due to its internal prompting. Conversely, when using an LLM, the plural form of the injected term is preserved as instructed by the prompt, but the surrounding elements, such as the article and verb, are not consistently adjusted to agree in number. In other words, sometimes verbs and prepositions do not agree their inflection according to the injected term.

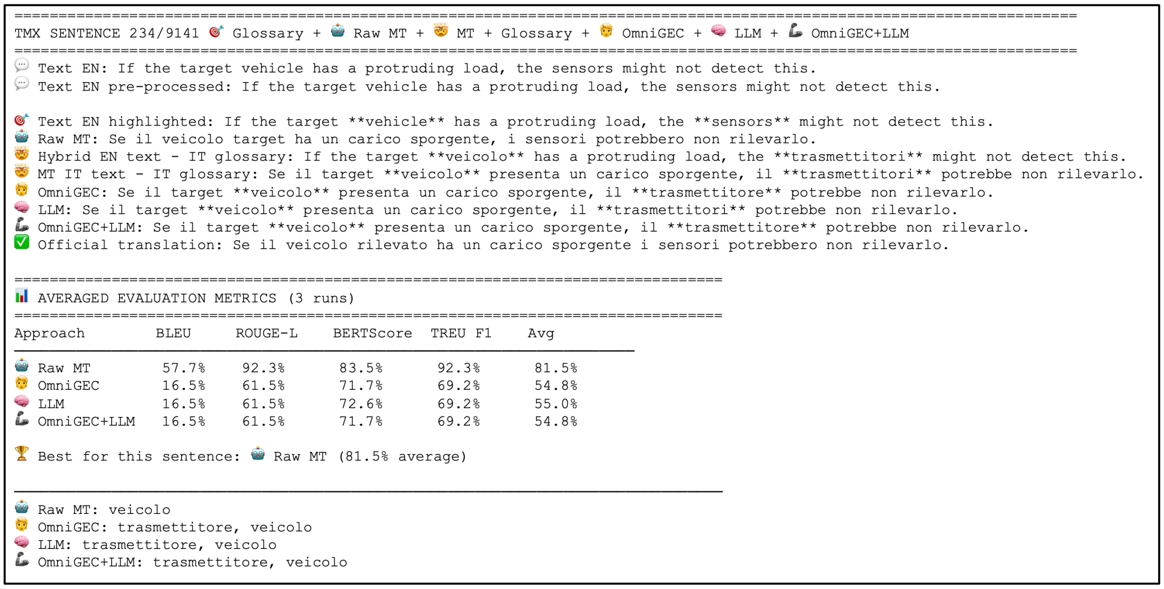
6. Insights about Volvo Terminology Dataset

As mentioned earlier, one of the major challenges concerns the handling of terminology injection. In this paper, we attempted to work with the terminology database as-is, without modifying its structure. However, to truly facilitate the overall process, the most effective approach would be to revise the database directly at the source and then adapt the code to it. Several improvements could substantially enhance consistency and accuracy:

1. Enriching POS, gender, number and usage context: adding part-of-speech tags, grammatical gender, singular/plural forms and usage context for each terminology entry would greatly simplify their retrieval and ensure correct agreement in Italian translations.

2. Including variants and synonyms: incorporating acceptable variants or synonyms would help resolve inconsistencies observed between official translations and prescribed terminology (e.g. the frequent use of "sensori" instead of "trasmettitori" as required by the database) (see Figure 7).

3. Improving multi-word term management: enhancing the handling of multi-word expressions, ensuring consistent English-Italian mappings and adding articles or prepositions when required, would markedly improve the grammatical integration of injected terms.



```
===========================================================================================================================
TMX SENTENCE 234/9141 🎯 Glossary + 🤖 Raw MT + 🤯 MT + Glossary + 👦 OmniGEC + 🧠 LLM + 🦾 OmniGEC+LLM
===========================================================================================================================
💬 Text EN: If the target vehicle has a protruding load, the sensors might not detect this.
💬 Text EN pre-processed: If the target vehicle has a protruding load, the sensors might not detect this.

🎯 Text EN highlighted: If the target **vehicle** has a protruding load, the **sensors** might not detect this.
🤖 Raw MT: Se il veicolo target ha un carico sporgente, i sensori potrebbero non rilevarlo.
🤯 Hybrid EN text - IT glossary: If the target **veicolo** has a protruding load, the **trasmettitori** might not detect this.
🤯 MT IT text - IT glossary: Se il target **veicolo** presenta un carico sporgente, il **trasmettitori** potrebbe non rilevarlo.
👦 OmniGEC: Se il target **veicolo** presenta un carico sporgente, il **trasmettitore** potrebbe non rilevarlo.
🧠 LLM: Se il target **veicolo** presenta un carico sporgente, il **trasmettitori** potrebbe non rilevarlo.
🦾 OmniGEC+LLM: Se il target **veicolo** presenta un carico sporgente, il **trasmettitore** potrebbe non rilevarlo.
✅ Official translation: Se il veicolo rilevato ha un carico sporgente i sensori potrebbero non rilevarlo.

==================================================================================
📊 AVERAGED EVALUATION METRICS (3 runs)
==================================================================================
Approach        BLEU     ROUGE-L     BERTScore  TREU F1     Avg
────────────────────────────────────────────────────────────────────────

🤖 Raw MT        57.7%     92.3%       83.5%       92.3%       81.5%
👦 OmniGEC       16.5%     61.5%       71.7%       69.2%       54.8%
🧠 LLM           16.5%     61.5%       72.6%       69.2%       55.0%
🦾 OmniGEC+LLM   16.5%     61.5%       71.7%       69.2%       54.8%

🏆 Best for this sentence: 🤖 Raw MT (81.5% average)

──────────────────────────────────────────────────────────────────────────────────
🤖 Raw MT: veicolo
👦 OmniGEC: trasmettitore, veicolo
🧠 LLM: trasmettitore, veicolo
🦾 OmniGEC+LLM: trasmettitore, veicolo
```

Figure 7: Example of inconsistency between the terminology used in the official translation and the one coming from the terminology database.

7. Conclusions and Future Work

This paper showed that combining terminology injection, machine translation, grammar correction models and LLM-based enhancement can occasionally improve translation quality in specialized technical domains, but still this combined pipeline in general produced worse translations compared to raw MT, which overall still provides the highest accuracy. We used different metrics and we found out that the most reliable between them for our terminology-sensitive translation task was TREU, even if with still some limitations. Several directions for future work emerge from our findings:

1. Fine-tuned MT models: instead of post-processing general-purpose MT systems, domain-specific MT models fine-tuned on relevant corpora may achieve better terminology and syntactic accuracy. Making a tuning on part of the TMX provided by Volvo would potentially increase considerably the accuracy of the translations.
2. Human evaluation: human assessment remains essential to validate terminology accuracy and to measure whether automated improvements translate to actual usefulness.
3. Improved syntax auto-correction: developing additional rule-based or statistical functions for resolving Italian agreement could mitigate recurring errors.

References

1. Dougal, D. K. & Lonsdale, D. W. (2020). “Improving NMT Quality Using Terminology Injection.”
2. Michon, E., Crego, J., & Senellart, J. (2020). “Integrating Domain Terminology into Neural Machine Translation.”
3. Chen, P. (2023). “Terminology-Aware Translation with Constrained Decoding and Large Language Model Prompting.”
4. Semenov, K. (2023). "Findings of the WMT 2023 Shared Task on Machine Translation with Terminologies.”
5. Lackner, C. (2024). “Review of Terminology Formats for In-Context Learning in LLMs.”

6. Papineni, K., Roukos, S., Ward, T., & Zhu, W.J. (2002). "BLEU: a Method for Automatic Evaluation of Machine Translation."
7. Chin-Yew Lin. (2004). "ROUGE: A Package for Automatic Evaluation of Summaries."
8. Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2019). "BERTScore: Evaluating Text Generation with BERT."
9. Kovalchuk, R., Romanyshyn, M., Ivaniuk, P. (2025). "Introducing OmniGEC: A Silver Multilingual Dataset for Grammatical Error Correction."

Requirements

- Excel terminology database extraction
- RAM: 8GB min, 16GB recommended | Storage: 16GB free space for models
- GPU: Optional, but highly recommended for OmniGEC model
- Python: 3.8 or higher | Library with spaCy, transformers, torch
- Ollama with LLaMA 3.1 or similar model | LibreTranslate server (both optional)