gApp-assisted NMT: how to improve the neural machine translation of discontinuous multiword expressions (IT→EN/DE)

Carlos Manuel Hidalgo-Ternero[1], Francesco Lista, and Gloria Corpas Pastor[2]

[1] *University of Malaga, IUITLM (Spain)*
[2] *University of Malaga, IUITLM (Spain)*

Abstract

This paper introduces gApp, a Python-based text preprocessing system designed for the automatic identification and conversion of discontinuous multiword expressions (MWEs) into their continuous form in order to improve neural machine translation (NMT). To evaluate its effectiveness, we conduct an experiment focusing on semi-fixed Verb + Prepositional Phrase (VPP) idioms and assess the extent to which gApp can enhance the performance of three NMT systems—ModernMT, Google Translate, and DeepL—when translating from Italian into English and German. The results show that the automatic conversion of discontinuous MWEs into their continuous forms leads to measurable improvements in translation quality across the evaluated systems. The paper concludes by discussing potential directions for further improving gApp-assisted NMT.

Keywords

text preprocessing system, Neural Machine Translation (NMT), Multiword Expression (MWE), Verb + Prepositional Phrase (VPP) idioms

1. Introduction

More than two decades after the seminal work of Sag and Baldwin [1], multiword expressions (MWEs) continue to pose major challenges for natural language processing systems, and machine translation is no exception. In addition to well-known difficulties such as syntactic irregularity, non-compositional meaning, and semantic ambiguity, another obstacle arises for neural machine translation (NMT): MWEs are not always realised as contiguous sequences of words (e.g., we took our invitation to the party for granted). Such discontinuous configurations significantly complicate their automatic detection and translation [2, 3].

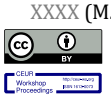
To address the difficulties that discontinuous MWEs continue to represent for even the most robust NMT systems, we developed gApp [4], a text pre-processing tool designed to automatically identify discontinuous MWEs and convert them into their continuous variants prior to translation. By normalising these expressions before they are processed by NMT engines, the system aims to facilitate their recognition and ultimately improve translation performance.

To demonstrate the effectiveness of gApp, seven different experiments have previously been carried out [4, 5, 6, 7, 8, 9, 10]. They comprise a total of 4,410 cases (1,470 discontinuous cases + 1,470 continuous ones after gApp + 1,470 continuous ones after the manual conversion), including different MWE typologies (verb-noun constructions and verb + prepositional phrase constructions), NMT

[1] SEPLN 2026: 42nd International Conference of the Spanish Society for Natural Language Processing, León, Spain, 22-25 September 2026.

email1@mail.com (A. Ometov); email2@mail.com (M. Y. Writter); email3@mail.com (J. X. Ceur)

XXXX-XXXX-XXXX-XXXX (A. Ometov); XXXX-XXXX-XXXX-XXXX (M. Y. Writter); XXXX-XXXX-XXXX-XXXX (J. X. Ceur)



systems (DeepL, Google Translate, and ModernMT), and translation directions (ES/FR→EN/DE/FR/ES/IT/PT/ZH). The global average results of Experiments 1–7 are presented in Table 1, which shows the NMT accuracy before gApp (column 1), the NMT accuracy after gApp (column 2), the improvement obtained with gApp (column 3), and the results of the gold standard, i.e., the manual conversion (column 4). Columns 3 and 4 are expressed in percentage points (pp).

Table 1
Global results from Experiments 1-7

Before gApp	After gApp	Improvement with gApp	Gold Standard
50.3%	65.4%	+15.1 pp	+15.8 pp

The global results presented in Table 1 show that gApp improved NMT performance by 15.1 pp, nearly reaching the gold standard improvement of 15.8 pp. Against this background, we have adapted gApp's detection and conversion mechanisms for Italian as the source language, in order to evaluate, for the first time, its performance with this newly implemented language. Within this study, we therefore aim to test our main hypothesis: that gApp can also enhance NMT performance by converting MWEs into their continuous forms in the IT→EN and IT→DE translation directions.

To this end, the remainder of the paper is structured as follows. Section 2 provides a brief overview of the relevant literature, followed by a description of the gApp system in Section 3. Section 4 outlines the research methodology, and Section 5 presents the evaluation of gApp's effectiveness in improving the selected NMT systems with respect to discontinuous MWEs. Section 6 discusses the results, and Section 7 concludes with remarks on potential strategies for further enhancing gApp-assisted NMT.

2. Related work

Improving the treatment of MWEs in MT has become an increasingly important research area. In the last decade, a substantial body of work has focused on MWE detection, leading to significant methodological advances [11, 12, 13, 14, among others]. Within this broader line of research, particular attention has been devoted to the automatic identification of discontinuous MWEs [3, 15, 16, 17, 18].

In parallel, several studies have also explored methods to improve the handling of MWEs within NMT architecture [19, 20, 21, 23]. In this context, several systems have been developed to support MWE processing using approaches comparable to those implemented in gApp. Examples include tools and frameworks proposed by Alegria et al. [24], Bejček, Straňák, and Pecina [16], Finlayson and Kulkarni [25], Nagy and Vincze [26], and Ramisch [27].

Many of these systems rely on token-based identification approaches, frequently implemented through the Lexicon Lookup method [28], whereby MWEs are detected in running text by matching tokens against a predefined inventory of lexical patterns. Some of these tools are also able to identify discontinuous expressions through mechanisms similar to those adopted in gApp. Such strategies include the use of gap-length constraints to limit the number of intervening tokens between the components of an expression [27], the implementation of surface realisation schemas (SRS) encoding information about the order of components, their possible continuity or discontinuity, and their inflectional constraints [24], and detection procedures requiring the simultaneous activation of both elements of a discontinuous expression to trigger its recognition [29], among other techniques.

Despite these developments, the preprocessing tool gApp represents a further step forward in comparison with existing approaches. Unlike previous systems, which are primarily designed to identify MWEs in text, gApp is not only able to detect discontinuous expressions but also capable of automatically transforming them into their continuous forms. As will be shown in the following sections, this additional functionality is particularly beneficial for NMT, since it facilitates the processing of such expressions and leads to measurable improvements in translation quality.

3. Overview of gApp

gApp was developed in Python 3.7 using the spaCy library, which provides a wide range of advanced natural language processing capabilities, including non-destructive tokenisation, part-of-speech tagging, dependency parsing, lemmatisation, and rule-based pattern matching, among others [30]. For this study, the pre-trained statistical model for Italian it_core_news_sm was employed. This model

is a multi-task convolutional neural network (CNN) trained on the WikiNER dataset [31] and the Universal Dependencies Italian ISDT dataset [32].

Within this framework, gApp was designed to carry out two primary preprocessing tasks for MWEs: (i) detecting discontinuous MWEs and (ii) converting them into continuous forms in order to enhance NMT performance.

3.1 Automatic detection of discontinuous MWEs

The system gApp adopts a token-based approach for identifying MWEs. This procedure follows the Lexicon Lookup method described by Ramisch and Villavicencio [28], whereby expressions are detected through comparison with a precompiled lexicon of semi-fixed MWEs. These MWEs allow a certain degree of morphosyntactic variation with respect to their canonical form. As a consequence, their components may appear separated by intervening elements in running text, producing discontinuous realisations.

Before designing the detection patterns used by the system, it was necessary to determine which types of n-grams could intervene between the components of these MWEs. To this end, concordances containing the discontinuous VPP (Verb + Prepositional Phrase) idioms were retrieved from the itTenTen20 corpus (14.5 billion tokens) and Italian Trends (2014–today) (12.3 billion tokens), both accessible through Sketch Engine. itTenTen20 is a large web corpus compiled from a wide range of Italian-language websites, including news sites, blogs, forums, and institutional pages, and therefore reflects diverse registers of contemporary written Italian. By contrast, Italian Trends (2014–today) corresponds to the JSI Timestamped corpus for Italian and mainly consists of continuously updated online news and media texts, providing a large collection of recent journalistic language [33].

Building on the corpus-based research methodology proposed by Hidalgo-Ternero [2], we used Sketch Engine's Corpus Query Language (CQL) schemas to extract both the discontinuous forms of the MWEs under study—hereafter referred to as *relevant results*—and other concordances with similar patterns but unrelated to the MWEs (*irrelevant results*). This process helps define the necessary constraints for gApp to optimise its precision and recall. The following example illustrates a CQL schema used to retrieve continuous and discontinuous (up to three tokens) concordances of the MWE *prendere di petto* ('to face something head on'):

```
[lemma="prendere"][]{0,3}[word="di"][word="petto"]
```

In this CQL schema, the verb *prendere* ('to take') was specified as a lemma in order to detect all of its conjugated forms, whereas the noun *petto* was defined as a fixed word, as it does not allow inflection within this MWE. The gap between the two elements was limited to a maximum of three tokens to increase precision, since the intervening material breaking the MWE sequence typically consisted of short noun phrases ([determiner] + noun + [adjective]), prepositional phrases (preposition + [determiner] + noun), or adverbial phrases.

Once the corpus concordances obtained with CQL schemas had been analysed, a set of rule-based matching patterns was incorporated into gApp's lexicon in order to capture as many relevant occurrences as possible while excluding irrelevant ones. Each pattern is represented as a list of dictionaries, which specify both the fixed components of the MWE and the elements that may intervene within the sequence. For instance, similarly to the CQL representation, in the MWE *prendere di petto*, the algorithm identifies the verb *prendere* through its lemma, allowing the detection process to capture all morphological variants. In contrast, the remaining elements (*di petto*) are treated as fixed tokens, since they can only occur in their canonical form for this MWE.

The corpus data also revealed that this MWE may be internally split by a variety of adverbial, nominal, or prepositional phrases, such as *il problema* ('the problem'), *la situazione* ('the situation'), *di nuovo* ('once again'), *finalmente* ('finally'), as well as by different subjects, among others. To account for these possibilities, the system restricts the first token appearing within the gap to a limited set of grammatical categories, namely determiners, adverbs, prepositions, pronouns, and proper nouns. Furthermore, tokens that may optionally appear inside the gap are marked using an optional attribute.

Finally, to minimise the number of irrelevant matches, the algorithm excludes cases in which the token immediately preceding the bigram *di petto* belongs to certain grammatical classes—

specifically verbs, coordinating conjunctions, subordinating conjunctions, or prepositions. Corpus concordances showed that sequences with these categories in that position typically correspond to structures unrelated to discontinuous realisations of the MWE, and therefore must be filtered out in order to increase the system's precision. Some stop words were also added within the gap for this MWE, such as *fetta* ('slice') or *tagliata* ('cut'), in order to filter out irrelevant results like the following one:

(1) **Prendete** delle fette **di petto** di pollo.
('Take some slices of chicken breast.')

In example 1, we observe an instance of the pattern "prendere (1-3 tokens) di petto", which however does not correspond to a discontinuous form of the MWE *prendere di petto*, since both *prendere* ('take') and *di petto* ('of breast') have a literal meaning in this context. For this reason, this instance constitutes an irrelevant result which must hence be filtered out (i.e., not converted) by gApp.

3.2 Automatic conversion of discontinuous MWEs

Once the detection phase has been completed, the system proceeds to a second stage in which discontinuous MWEs are automatically transformed into their continuous variants. This procedure is implemented through a loop that first verifies whether any of the predefined patterns stored in the MWE lexicon are present in the analysed text.

When a pattern is detected, the algorithm determines the initial position of the match (*pos_ini*) and the final position (*pos_fin*) corresponding to the first and last elements of the MWE. The intervening sequence between these two positions is treated as the gap, whose elements are optional within the expression. The first token inside the gap (gap1) is assigned the position *pos_ini + 1*, while the last token (gap3) corresponds to *pos_fin − 3*.
After identifying the relevant positions, the algorithm reconstructs the sentence by concatenating several segments:

1. the portion of text from the beginning of the sentence up to and including *pos_ini*;
2. the token corresponding to *pos_fin*;
3. the sequence of tokens from *gap1* to *gap3*;
4. the remaining part of the sentence following *pos_fin*.

Through this reordering procedure, the discontinuous expression is reassembled into its continuous form in the final output. If the system does not detect any of the patterns defined in the lexicon, the input text remains unchanged. The procedure is applied iteratively until all discontinuous MWEs identified in the text have been converted.

Once the system had been configured for Italian, we calculated gApp's precision and recall for the MWEs under study. To this end, a total of 400 relevant results (constituting our dataset) and 400 irrelevant results were compiled for processing by gApp.

Regarding gApp's ability to filter in as many relevant results while filtering out as many irrelevant ones as possible, gApp attained a precision of 98% and a recall of 95.1%, resulting in an F2 score of 95.7%. This means that only 2% of converted cases corresponded to irrelevant results, and only 4.9% of the relevant results were missed.

4. Methodology

This section presents the research methodology implemented to assess whether gApp can improve the performance of the NMT systems ModernMT, Google Translate and DeepL in the IT→EN and IT→DE translation directions. The methodology is structured into three main subsections: (4.1) the MWEs under study, (4.2) the selection of NMT systems and dataset construction, and (4.3) the NMT evaluation process.

4.1 The MWEs under study

In order to test gApp's effectiveness in the IT→EN/DE translation directions, the following Italian VPP idioms were selected for this study:

1) *prendere di petto* (with a frequency of 0.32 per million tokens [pmt] in itTenTen20): literally, 'to take something of heart'; figuratively, 'to face something head on'.[2]

[2] These English translations of the MWEs were retrieved from WordReference [49].

2) *vedere di buon occhio* (0.64 pmt): lit., 'to see something of good eye'; fig., 'to see something in a good light'.
3) *ringraziare di cuore* (1.46 pmt): lit., 'to thank something of heart'; fig., 'to thank something from the bottom of one's heart'.
4) *avere per le mani* (0.43 pmt): lit., 'to have something around the hands'; figuratively, 'to deal with something'.
5) *prendere per il naso* (0.1 pmt): lit., 'to take someone by the nose'; fig., 'to make a fool of someone'.
6) *tenere d'occhio* (3.11 pmt): lit., 'to have something/someone of eye'; fig., 'to keep an eye on something/someone'.
7) *andare di corpo* (0.16 pmt): lit., 'to walk of body'; fig., 'to defecate'.
8) *legarsela al dito* (0.08 pmt): lit., 'to stick something to your finger'; fig., 'to bear a grudge', 'to hold something against someone'.

These MWEs were chosen because they exhibit a range of linguistic properties, including idiomaticity, cultural specificity, and the potential for both continuous and discontinuous forms. We specifically selected somatisms (i.e., lexemes referring to human or animal body parts), because they represent a highly productive domain within phraseology and are widely attested across languages, often drawing on shared patterns of embodied metaphor. Their relatively high degree of conventionalisation and the frequent presence of idiomatic correspondences across languages make them especially suitable for controlled comparative analysis, while also facilitating expert evaluation of translation adequacy [34, 35].

Regarding the specific category of these MWEs, following Ramisch's MWE taxonomy [27], they can be classified as *idiomatic expressions*, since they have a non-compositional meaning (and are therefore also defined as *semantically non-decomposable idioms* or *SNDIs* [35]). With regard to the frequency of appearance of their different realisations (continuous vs. discontinuous), the concordances obtained from the itTenTen20 corpus show that, on average, these MWEs mostly appear in their continuous form (81% of continuous appearances vs. 19% discontinuous ones). Among them, the MWE *avere per le mani* shows a considerably lower percentage of continuous occurrences, although these still double the proportion of discontinuous ones (66.8% vs. 33.2%).

4.2 Selection of NMT systems and dataset construction

Regarding the MT systems under evaluation, the following three NMTs were selected: Google Translate, DeepL and ModernMT. Google Translate was chosen because it is one of the most widely used NMT systems worldwide [36]. DeepL was also included as one of the highest-performing NMT systems currently available [37]. Since the DeepL API offers the option to choose between the "classic language model" (NMT, according to the company) and the "next-gen language model" (a translation-specific LLM), we opted for the classic model in order to evaluate its NMT capabilities more specifically in this study. ModernMT was included as a research-oriented, adaptive NMT system designed to learn from domain-specific data and to provide high-quality translations across a variety of languages and contexts [38, 39], thus offering a complementary perspective to the other two MT systems. The translations from Google Translate, DeepL, and ModernMT were generated through their commercial APIs, with each text submitted individually to ensure that the systems considered only the context of each specific example.

Concerning the dataset, a variety of text sources, types, and language varieties was retrieved from the previously described corpora itTenTen20 and Italian Trends (2014–today). In this regard, despite the challenges posed by user-generated content (UGC)—including source-text errors, noise, and out-of-vocabulary tokens—even for the most robust NMT systems [40, 41], concordances containing UGC were also included in the analysis in order to mitigate potential sampling bias that might arise from examining only canonical training data for these MWEs. The selected texts did not only consist of the idiom by itself, but also the preceding and following sentences, so that the MT systems can have the context for translation.

The texts containing discontinuous instances of the MWEs under study were then converted into their continuous forms both automatically by gApp and manually. These texts were pasted into a spreadsheet in 3 columns: (i) before gApp, (ii) after the automatic conversion with gApp, and (iii) after the manual conversion. The NMT outputs for these source texts were also pasted into the same spreadsheet ordered by NMT system (ModernMT,

Google Translate, and DeepL) and by translation direction (IT→EN and IT→DE), thus creating the complete dataset.

In this way, a total of 1,200 texts were analysed, comprising 400 containing the discontinuous forms of the MWEs, 400 containing their continuous forms after gApp, and 400 containing their continuous form after the manual conversion. Our complete dataset hence comprises 1,200 texts × 3 NMT systems × 2 language directions = 7,200 translated texts.

4.3 NMT evaluation process

For the evaluation process, three professional Italian-to-English/German translators with more than four years of professional experience were recruited to manually assess the translations. Manual evaluation was chosen due to the challenges that automatic metrics face in accurately assessing MWE translation, especially since they tend to evaluate the translation as a whole and, consequently, the specific rendering of idioms in the target text is often overlooked [42].

The evaluators used a binary scale (1 = good, 0 = bad) to rate the quality of each translation, focusing exclusively on the accurate rendering of the MWEs in the target text. The evaluators were instructed to pay particular attention to the translation of discontinuous MWEs, assessing whether the intervening elements were appropriately handled and whether the idiomatic meaning was preserved.

If the system identified and adequately translated the Italian MWE into a valid English form, the score was 1 (e.g., "Loro non vedono la proposta di buon occhio." → "They don't approve the proposal."); otherwise, the score was 0 (e.g., "Loro non vedono la proposta di buon occhio." → "They do not see the proposal of good eye.").

For MWEs in their discontinuous form, only the MT outputs that identified the Italian MWE and appropriately translated it into an adequate English/German form, while preserving the intervening element, were scored as 1 (e.g., "L'hanno preso un'altra volta per il naso." → "They fooled him once again."). If the MT systems adequately translated an MWE but omitted the intervening element, the translation was considered incorrect (score 0) (e.g., "L'hanno preso un'altra volta per il naso." → "They fooled him.", omitting the intervening element "un'altra volta"/"once again").

As for inter-annotator agreement, the final decision was based on a majority rule: when two or three evaluators rated a translation as 0 or 1, that rating was adopted as the final score. To verify the reliability of the human evaluation, inter-annotator agreement was assessed using Cohen's kappa, which resulted in a score of 1 [43]. This indicates complete agreement between the three reviewers across all dataset examples. Although such full agreement is uncommon in machine translation evaluation frameworks such as MQM or adequacy/fluency assessments [44], it can be more readily achieved when using a binary rating scale.

5. Results

Regarding the IT→EN translation direction, Figure 1 illustrates the global results for the different NMT systems under analysis. In absolute terms, Google Translate is the system that provides the best results for the continuous forms of the MWEs after the automatic conversion with gApp, reaching 73.5% of appropriate outputs, compared to 32.5% with ModernMT and 60.5% with DeepL. A similar tendency can be observed after the manual conversion, where Google Translate achieves 73.8%, whereas ModernMT reaches 33.3% and DeepL reaches 60.5%. In terms of improvement between the initial results and the converted forms, Google Translate again shows the largest gain, increasing from 61.5% before conversion to 73.5% after the automatic conversion (+12 percentage points [pp]) and 73.8% after the manual conversion (+12.3 pp). ModernMT also displays a substantial improvement, rising from 22.5% to 32.5% after the automatic conversion (+10 pp) and 33.3% after the manual conversion (+10.8 pp). By contrast, DeepL shows a more limited variation, increasing from 56.8% to 60.5% after both the automatic and manual conversions (+3.7 pp).

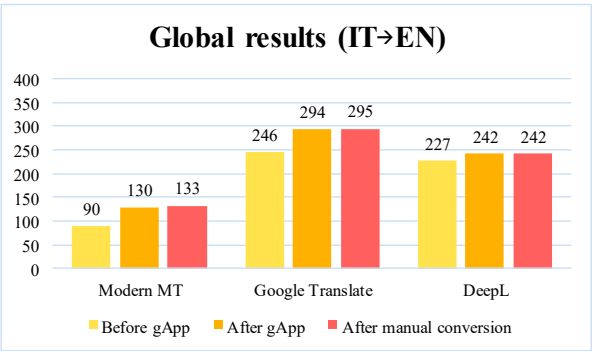


Figure 1: Global results (IT→EN)

Figure 2 illustrates the global results regarding the IT→DE translation direction. In this case as well, Google Translate achieves the highest scores among the evaluated systems. Its performance increases from 53% appropriate outputs before conversion to 61.3% after the automatic conversion with gApp (+8.3 pp) and 61.5% after the manual conversion (+8.5 pp). ModernMT also shows a noticeable improvement, rising from 25% before conversion to 30.8% after the automatic conversion (+5.8 pp) and 31.3% after the manual conversion (+6.3 pp). Meanwhile, DeepL records a more moderate increase, moving from 52% before conversion to 54.8% after both the automatic and manual conversions (+2.8 pp). Overall, although all systems benefit from the conversion process, Google Translate once again shows the largest relative improvement, followed by ModernMT, whereas DeepL presents smaller gains.

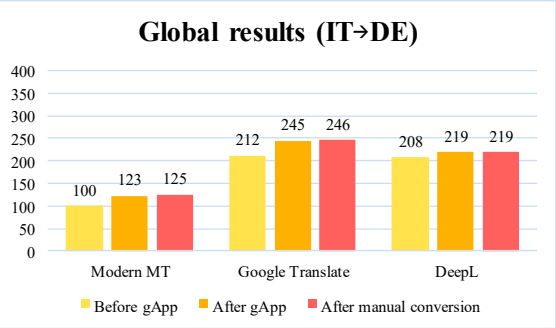


Figure 2: Global results (IT→DE)

When all systems and both translation directions are considered together, the overall tendency becomes even more evident. Out of a total of 2,400 evaluated outputs, the number of appropriate translations rises from 1,083 (45.1%) with the original discontinuous forms to 1,253 (52.2%) after the automatic conversion with gApp, and 1,260 (52.5%) after the manual conversion. In other words, the automatic conversion alone leads to an improvement of 7.1 pp, while the manually converted version yields a slightly higher gain of 7.4 pp.

6. Discussion

Global results with ModernMT, Google Translate and DeepL in the IT→EN/DE directions suggest that converting discontinuous idiomatic expressions into continuous forms before translation can significantly facilitate their processing by these NMT systems. In this context, across both translation directions, Google Translate consistently achieves the highest gains after conversion, followed by ModernMT, whereas DeepL delivers more modest improvements.

Global results also show that gApp managed to deliver a remarkable performance for the analysed MWEs, which was only 0.3 pp lower than the manual conversion. This is chiefly due to gApp's refined detection system both in terms of final average precision (98%) and recall (95.1%), which means that only 2% of the irrelevant results could enter the system and only 4.9% of the relevant results were not successfully detected.

Some of the most difficult instances for gApp to identify were, on the one hand, sentences containing rare or out-of-vocabulary (OOV) items, which often resulted in incorrect parsing. For example, in *ho un real wedding per le mani*, the word *real* was analysed as a verb rather than as an adjective, since the English sequence *real wedding* is uncommon in Italian. A similar problem occurs in *avere i dvd yamato per le mani*, where *yamato* was incorrectly tagged as a verb instead of a noun.

Another group of problematic cases involved structurally ambiguous sequences that allowed multiple syntactic or semantic interpretations. An illustrative example is *hanno l'oro colato per le mani*. At first sight, this sequence may be interpreted either as *hanno l'oro colato per le mani*, where *colato* functions as an adjective modifying the noun phrase *l'oro*, or as *hanno l'oro colato per le mani*, where *colato* is analysed as a past participle modified by the prepositional phrase *per le mani*. The latter interpretation is the one selected by gApp. However, a comparison of corpus frequencies suggests that this analysis is unlikely. When the expressions *oro colato* and *colare [0–3 tokens] per le mani* are queried in the itTenTen20 corpus, the former retrieves 6,764 concordances (0.47 per million words), whereas the latter only yields two relevant occurrences, one of them being the example discussed here (*hanno l'oro colato per le mani*). This indicates that *oro colato* should be treated as a MWE embedded within another MWE, namely *avere per le mani*.

A similar misinterpretation occurs in the sentence *ho un testo commissionato per le mani*. In this case, gApp parses the structure as *ho un testo commissionato per le mani* whereas the correct interpretation would be *ho un testo commissionato per le mani*. These examples highlight the need to further refine gApp's POS tagging and syntactic parsing components, particularly when dealing with non-canonical user-generated content and rare

lexical items [45, 46, 47], and with concordances that allow multiple syntactic or semantic readings.

The overall results indeed show that gApp produces substantial relative improvements when the performance obtained with discontinuous expressions is compared with that achieved after converting them into continuous forms. However, the absolute scores reveal that current NMT systems still require further development to reach satisfactory levels of performance, suggesting that discontinuity represents only one of several challenges involved in translating such expressions.

In fact, in the IT→EN translation direction, even after the conversion performed by gApp, ModernMT achieves only 32.5% appropriate outputs, while Google Translate reaches 73.5% and DeepL 60.5%. A comparable pattern emerges for the IT→DE direction: despite the improvements resulting from the conversion process, ModernMT only attains 30.8% appropriate translations, whereas Google Translate and DeepL reach 61.3% and 54.8%, respectively. These findings suggest that, although transforming discontinuous expressions into continuous ones can facilitate translation, a considerable number of cases remain problematic for current NMT systems.

A closer examination of the selected somatisms sheds further light on the main translation difficulties. In the case of *prendere di petto*, within the IT→EN translation direction, ModernMT provides only four adequate translations. In most instances, the system misinterprets both discontinuous and continuous variants of the MWE, rendering it literally as *to take on someone's chest* or by incorrectly analysing *chest* or *breast* as modifiers of the preceding noun phrase (e.g., *I've taken breast studies*; *taking the politics of the chest*; *take the initiative of chest*).

Other problematic outputs for this MWE include translations such as *to take (something) by heart*, which in English normally means 'to memorise', or *to take (something) to heart*¸ i.e., 'to take something seriously and be affected or upset by it' [48]. In these cases, the system appears to associate the Italian word *petto* ('chest') with *heart*, presumably due to their anatomical proximity. Despite the generally poor absolute results produced by ModernMT, Tables 2 (source texts [STs]) and 3 (target texts [TTs]) provide an example in which the conversion carried out by gApp leads to a noticeable improvement: after preprocessing, the system generates a considerably more appropriate translation than in the original discontinuous version (the entire sequence is highlighted in bold and the somatism is underlined).

Table 2
KWIC extracts for *prendere di petto* before and after gApp.

	KWIC extracts
ST (IT) before gApp	[...] Ho la netta sensazione che i dirigenti del PD abbiano paura di **<u>prendere</u> la situazione <u>di petto</u>** ed incominciare a far saltare tutto. […] [3]
ST (IT) after gApp	[...] Ho la netta sensazione che i dirigenti del PD abbiano paura di **<u>prendere di petto</u> la situazione** ed incominciare a far saltare tutto. [...]

Table 3
ModernMT's results for *prendere di petto* before and after gApp

	ModernMT
TT (EN) before gApp	[…] I have a distinct feeling that PD leaders are afraid to **<u>take</u> the situation <u>to heart</u>** and start blowing it all up. […]
TT (EN) after gApp	[…] I have a distinct feeling that PD leaders are afraid to **<u>take</u> the situation <u>head-on</u>** and start blowing it all up. […]

The IT→DE translation direction offers additional insights. As observed in previous studies [7], translations into German are often strongly influenced by English. Tables 4 and 5 illustrate such a situation with the MWE *prendere di petto*. In the example analysed, the continuous expression *prendere di petto la vita* is first mistranslated into English as *take life by heart*, and subsequently rendered into German as *das Leben auswendig zu nehmen* (literally, 'to take life by heart', i.e. 'to take life from memory'). The only connection between

[3] Due to space constraints, the tables include only the KWIC extracts and their TTs. The remaining text has been omitted here and replaced with "[...]" for illustrative purposes.

the Italian source and the German output lies in the English intermediary translation, which links *petto* ('chest') with *heart*, thereby producing a completely different expression. The final translation into German can hence only be explained by pivoting through English:

prendere di petto la vita → *to take life by heart* (*heart* is semantically related to *petto* ['chest'] in Italian) → *das Leben auswendig zu nehmen* (*auswendig* is semantically related to *by heart* [i.e., 'from memory'] in English)

Table 4
KWIC extract for *prendere di petto* after gApp.

	KWIC extract
ST (IT) after gApp	[…] Scrivo per la prima volta in questo magnifico quaderno, la storia se mi concedete il termine, di una donna, che ha deciso di **prendere di petto la vita** e di affrontarla, [...]

Table 5
ModernMT's results for *prendere di petto* after gApp.

	ModernMT
TT (EN)	[…] the story if you will grant me the term, of a woman, who decided to **take life by heart** and to face it, […]
TT (DE)	[…] die Geschichte, wenn Sie mir den Begriff einer Frau geben, die beschlossen hat, **das Leben auswendig zu nehmen** und sich ihm zu stellen, [...]

These mistranslations therefore emphasise the necessity for more training data in languages different from English, in order to avoid English-centred NMT outcomes.

Another particularly challenging expression for all three systems was *avere per le mani*, especially in the IT→DE translation direction. Several factors may account for this difficulty. In Italian, this expression is typically used idiomatically, meaning 'to deal with something at the moment' or 'to have something underway' (e.g., *ho una faccenda per le mani*; *ho un problema per le mani*; *ho un affare per le mani*). However, NMT systems frequently interpreted it literally, translating it as 'to have something in one's hands'. In addition, the systems may confuse *avere per le mani* with the related variant *avere tra le mani*. Depending on the context, this variant can share the same idiomatic meaning but is also commonly used in a literal sense ('to hold something in one's hands'), for instance in expressions such as *ho il tuo viso fra le mani*, *ho della legna fra le mani*, or *ho un cucciolo fra le mani*. This overlap between literal and figurative meanings often leads to inadequate translations.

In other cases, *avere per le mani* was misinterpreted through a completely different MWE. Tables 6 and 7 illustrate a translation provided by DeepL, in the IT→DE translation direction, where the MWE *abbia un buon colpo per le mani* is mistranslated as *einen guten Treffer gelandet hat*. In this regard, DeepL translates *colpo* as *Treffer* (which could be a quite acceptable translation in other contexts), but it completely changes the context and adapts the word *Treffer* to an already existing idiom in German, i.e., *einen Treffer landen*, meaning 'to be successful' (which, in Italian, would be *fare centro*). Moreover, the discontinuous form leads to an inappropriate tense shift in the translation: *abbia* is in the present subjunctive, whereas *gelandet hat* is in the past tense (Perfekt), which is incorrect since that coup had not taken place yet. Against this backdrop, gApp helps DeepL produce a much more appropriate translation (*die einen guten Coup in Aussicht hat*, i.e., 'who has a good coup in prospect'), as shown in Tables 6 and 7.

Table 6
KWIC extracts for *avere per le mani* before and after gApp.

	KWIC extracts
ST [IT] (before gApp)	[...] Ma Driver all'occasione accetta anche di fare da autista per una gang che **abbia un buon colpo per le mani** . [..]
ST [IT] (after gApp)	[...] Ma Driver all'occasione accetta anche di fare da autista per una gang che **abbia per le mani un buon colpo**. [...]

Table 7
DeepL's results for *avere per le mani* before and after gApp.

	DeepL
TT [DE] (before gApp)	[…] Aber Driver erklärt sich auch bereit, als Fahrer für eine Bande zu arbeiten, die **einen guten Treffer gelandet hat**. [...]
TT [DE] (after gApp)	[…] Aber Driver erklärt sich auch bereit, als Fahrer für eine Bande zu arbeiten, die **einen guten Coup in Aussicht hat**. [...]

7. Conclusion

Our findings confirm the initial hypothesis that gApp enhances NMT quality for discontinuous MWEs by converting them into continuous forms. In this regard, the average improvement after gApp was 7.2 pp for IT→EN and 6.9 pp for IT→DE. Across both language pairs, the overall average gain reached 7.1 pp with gApp (vs. 7.4 pp with the gold standard), remaining below the improvements observed in Experiments 1–7 (15.1 pp with gApp vs. 15.8 pp with the gold standard, see Table 1). This lower performance may stem from the inclusion of a new source language (Italian), though further research is needed to confirm this.

The results of this and previous experiments highlight the potential of gApp as an effective preprocessing tool for discontinuous MWEs. In light of these encouraging outcomes, future research could explore scaling up this approach to other NLP tasks where pre-processing of discontinuous MWEs may play a crucial role, such as information retrieval, automatic text summarisation, sentiment analysis, and beyond. Although the present study offers meaningful insights, several limitations point to potential avenues for future research. First, the analysis was restricted to the Italian–English and Italian–German translation directions. Subsequent studies could broaden the scope to additional language pairs, especially those involving low-resource languages, to examine the extent to which the results can be generalised or transferred to other linguistic contexts. Furthermore, the MWEs considered here mainly consist of general-language idioms. Future research could therefore investigate the translation of domain-specific MWEs, such as those occurring in legal or technical discourse, to determine how MT systems perform in more specialised domains. Finally, another limitation concerns the selection of Google Translate, DeepL, and ModernMT as representative NMT systems. While their performance offers useful insights into the current capabilities of neural MT, the reliance on proprietary systems may restrict the reproducibility and transparency of the study. Future work will therefore incorporate open-source NMT models in order to promote greater research transparency and facilitate wider access to the tools and procedures employed.

Regarding gApp, several constraints have been identified throughout this study that point to the need for further improvements to its POS tagger and syntactic parser, in order to handle rare and out-of-vocabulary words as well as potentially ambiguous sentences more effectively. In addition, the construction of its lexicon currently requires extensive manual effort and depends on language-specific heuristics, which do not generalise well across different languages, MWE typologies, or domains. Consequently, future research will explore whether state-of-the-art Large Language Models (LLMs) can assist in expanding gApp's lexicon to other MWEs and languages, and whether gApp can also enhance machine translation produced by LLMs. These are, therefore, some of the challenges that still lie ahead of us in the relentless race to bridge the "gApp" in NMT quality.

Acknowledgements

This research was carried out within the framework of several research projects (ref. PPRO-IUI-2023-06, PID2024-160929OB-I00/ MICIU/AEI/10.13039/501100011033/, HUM106-GFEDER, and DGP_PIDI_2024_01159) at the University of Malaga and at the Research Institute of Multilingual Language Technologies (IUITLM) (Spain).

References

[1] Sag, I. A., Baldwin, T., Bond, F., Copestake, A., Flickinger, D., Multiword expressions: A pain in the neck for NLP, in: A. Gelbukh (Ed.), Computational Linguistics and Intelligent Text Processing, Springer, 2002, pp. 1–15.

[2] C. M. Hidalgo-Ternero, Google Translate vs. DeepL: Analysing neural machine translation performance under the challenge of phraseological variation, MonTI Special Issue 6 (2020) 154–177. doi:10.6035/MonTI.2020.ne6.5.

[3] O. Rohanian, S. Taslimipoor, S. Kouchaki, L. An Ha, R. Mitkov, Bridging the gap: Attending to discontinuity in identification of multiword expressions, in: J. Burstein, C. Doran, T. Solorio (Eds.), Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, Vol. 1, Association for Computational Linguistics, 2019, pp. 2692–2698.

[4] C. M. Hidalgo-Ternero, G. Corpas Pastor, Bridging the 'gApp': improving neural machine translation systems for multiword expression detection, Yearbook of Phraseology 11 (2020) 61–80. doi:10.1515/phras-2020-0005.

[5] C. M. Hidalgo-Ternero, El algoritmo ReGap para la mejora de la traducción automática neuronal de expresiones pluriverbales discontinuas (FR>EN/ES), in: G. Corpas Pastor, M. R. Bautista Zambrana, C. M. Hidalgo-Ternero (Eds.), Sistemas Fraseológicos en Contraste: Enfoques Computacionales y de Corpus, Comares, 2021, pp. 253–270.

[6] C. M. Hidalgo-Ternero, ¿DeepL, Google Translate o VIP? Qué sistema ofrece un mejor rendimiento en la traducción de locuciones continuas y discontinuas, in: G. Corpas Pastor, F. J. Veredas Navarro (Eds.), Tecnologías Lingüísticas Multilingües: Desarrollos Actuales y Transición Digital, Comares, 2024.

[7] C. M. Hidalgo-Ternero, G. Corpas Pastor, ReGap: A text preprocessing algorithm to enhance MWE-aware neural machine translation systems, in: J. Monti, G. Corpas Pastor, R. Mitkov, C. M. Hidalgo-Ternero (Eds.), Recent Advances in MWU in Machine Translation and Translation Technology, John Benjamins Publishing Company, 2024. doi:10.1075/cilt.366.02hid.

[8] C. M. Hidalgo-Ternero, G. Corpas Pastor, Qué se traerá gApp entre manos… O cómo mejorar la traducción automática neuronal de variantes somáticas (ES>EN/DE/FR/IT/PT), in: M. Seghiri, M. Pérez Carrasco (Eds.), Nuevos Enfoques en Traducción Científica, Técnica y Agroalimentaria, Peter Lang, 2026.

[9] C. M. Hidalgo-Ternero, X. Zhou Lian, Reassessing gApp: Does MWE discontinuity always pose a challenge to neural machine translation?, in: G. Corpas Pastor, R. Mitkov (Eds.), Computational and Corpus-Based Phraseology, Springer Nature Switzerland, 2022, pp. 116–132.

[10] C. M. Hidalgo-Ternero, X. Zhou-Lian, Minding the "gApp" in the neural machine translation of discontinuous idioms (ES>EN/ZH), Estudios Interlingüísticos 13 (2025) 111–129.

[11] N. Klyueva, A. Doucet, M. Straka, Neural networks for multi word expression detection, in: Proceedings of the 13th Workshop on Multiword Expressions, 2017, pp. 60–65. doi:10.18653/v1/W17-1707.

[12] A. Maldonado, L. Han, E. Moreau, A. Alsulaimani, K. D. Chowdhury, C. Vogel, Q. Liu, Detection of verbal multi word expressions via conditional random fields with syntactic dependency features and semantic re ranking, in: Proceedings of the 13th Workshop on Multiword Expressions, 2017, pp. 114–120. doi:10.18653/v1/W17-1715.

[13] M. Riedl, C. Biemann, Impact of MWE resources on multiword recognition, in: Proceedings of the ACL 2016 Workshop on Multiword Expressions, 2016, pp. 107–111.

[14] N. Zampieri, C. Ramisch, G. Damnati, The impact of word representations on sequential neural MWE identification, in: Joint Workshop on Multiword Expressions and WordNet (MWE-WN 2019), 2019, pp. 169–175. doi:10.18653/v1/W19-5121.

[15] H. Al Saied, M. Candito, M. Constant, Comparing linear and neural models for

competitive MWE identification, in: Proceedings of the 22nd Nordic Conference on Computational Linguistics, 2019, pp. 86–96.
[16] E. Bejček, P. Straňák, P. Pecina, Syntactic identification of occurrences of multiword expressions in text using a lexicon with dependency structures, in: Proceedings of the 9th Workshop on Multiword Expressions, 2013, pp. 106–115.
[17] M. Constant, G. Eryiğit, J. Monti, L. van der Plas, C. Ramisch, M. Rosner, A. Todirascu, Multiword expression processing: A survey, Computational Linguistics 43 (2017) 1–92. doi:10.1162/COLI_a_00302.
[18] E. Moreau, A. Alsulaimani, A. Maldonado, C. Vogel, CRF Seq and CRFDepTree at PARSEME Shared Task 2018: Detecting verbal MWEs using sequential and dependency based approaches, in: Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG 2018), 2018, pp. 241–247.
[19] N. Schneider, E. Danchik, C. Dyer, N. A. Smith, Discriminative lexical semantic segmentation with gaps: Running the MWE gamut, Transactions of the Association for Computational Linguistics 2 (2014) 193–206.
[20] P. S. Huang, C. Wang, S. Huang, D. Zhou, L. Deng, Towards neural phrase based machine translation, arXiv preprint arXiv:1706.05565, 2018. URL: https://arxiv.org/pdf/1706.05565.
[21] M. Rikters, O. Bojar, Paying attention to multi word expressions in neural machine translation, arXiv preprint arXiv:1710.06313, 2017.
[22] X. Wang, Z. Tu, D. Xiong, M. Zhang, Translating phrases in neural machine translation, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017, pp. 1421–1431. doi:10.18653/v1/D17-1149.
[23] A. Zaninello, A. Birch, Multiword expression aware neural machine translation, in: Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), 2020, pp. 3816–3825.
[24] I. Alegria, O. Ansa, X. Artola, N. Ezeiza, K. Gojenola, R. Urizar, Representation and treatment of multiword expressions in Basque, in: Proceedings of the Second ACL Workshop on Multiword Expressions: Integrating Processing, 2004, pp. 48–55.
[25] M. Finlayson, N. Kulkarni, Detecting multiword expressions improves word sense disambiguation, in: Proceedings of the ALC Workshop on MWEs (MWE 2011), 2011, pp. 20–24.
[26] T. Nagy, V. Vincze, VPCTagger: Detecting verb particle constructions with syntax based methods, in: Proceedings of the 10th Workshop on Multiword Expressions, Association for Computational Linguistics, 2014.
[27] C. Ramisch, Multiword Expressions Acquisition: A Generic and Open Framework, Springer, 2015.
[28] C. Ramisch, A. Villavicencio, Computational treatment of multiword expressions, in: R. Mitkov (Ed.), Oxford Handbook on Computational Linguistics, 2nd ed.
[29] V. Foufi, L. Nerima, E. Wehrli, Multilingual parsing and MWE detection, in: Y. Parmentier, J. Waszczuk (Eds.), Representation and Parsing of Multiword Expressions: Current Trends, Language Science Press, 2019, pp. 217–237.
[30] M. Honnibal, I. Montani, S. Van Landeghem, A. Boyd, spaCy 2: Natural Language Understanding with Bloom Embeddings, Convolutional Neural Networks and Incremental Parsing, Software documentation, Explosion AI, 2020. URL: https://spacy.io/.
[31] J. Nothman, N. Ringland, W. Radford, T. Murphy, J. R. Curran, Learning multilingual named entity recognition from Wikipedia, Figshare dataset, 2013. doi:10.1016/j.artint.2012.03.006.

[32] C. Bosco, F. Dell'Orletta, S. Montemagni, M. Sanguinetti, M. Simi, The Evalita 2014 dependency parsing task, in: R. Basili, A. Lenci, B. Magnini (Eds.), Proceedings of the First Italian Conference on Computational Linguistics CLiC-it 2014 & Fourth International Workshop EVALITA 2014, Pisa University Press, 2014, pp. 1–8.

[33] A. Kilgarriff, P. Rychly, P. Smrz, D. Tugwell, The Sketch Engine, in: Proceedings of the 11th EURALEX International Congress, 2004, pp. 105–116.

[34] C. Mellado Blanco, Fraseologismos somáticos del alemán, Peter Lang, 2004.

[35] S. Bargmann, M. Sailer, The syntactic flexibility of semantically non-decomposable idioms, in: M. Sailer, S. Markantonatou (Eds.), Multiword Expressions: Insights from a Multi-Lingual Perspective, Language Science Press, 2018, pp. 1–29.

[36] I. Rivera-Trigueros, Machine translation systems and quality assessment: A systematic review, Language Resources and Evaluation 56 (2022) 593–619. doi:10.1007/s10579-021-09537-5.

[37] M. I. Kamaluddin, M. W. K. Rasyid, F. H. Abqoriyyah, A. Saehu, Accuracy analysis of DeepL: Breakthroughs in machine translation technology, Journal of English Education Forum (JEEF) 4 (2024) 122–126.

[38] N. Bertoldi, D. Caroselli, M. Federico, The ModernMT Project, in: Proceedings of the 21st Annual Conference of the European Association for Machine Translation, Alicante, Spain, 2018, p. 365.

[39] L. Macken, V. De Wilde, A. Tezcan, Machine translation for open scholarly communication: Examining the relationship between translation quality and reading effort, Information 15 (2024) 427. doi:10.3390/info15080427.

[40] Y. Belinkov, Y. Bisk, Synthetic and natural noise both break neural machine translation, arXiv preprint arXiv:1711.02173, 2018. URL: https://arxiv.org/abs/1711.02173.

[41] P. Lohar, M. Popović, H. Alfi, A. Way, A systematic comparison between SMT and NMT on translating user generated content, in: Proceedings of the 20th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing 2019), 2019.

[42] C. Baziotis, P. Mathur, E. Hasler, Automatic evaluation and analysis of idioms in neural machine translation, in: A. Vlachos, I. Augenstein (Eds.), Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, Dubrovnik, Croatia, 2023, pp. 3682–3700. doi:10.18653/v1/2023.eacl-main.267.

[43] R. Artstein, Inter-annotator agreement, in: N. Ide, J. Pustejovsky (Eds.), Handbook of Linguistic Annotation, Springer Netherlands, Dordrecht, 2017, pp. 297–313. doi:10.1007/978-94-024-0881-2_11.

[44] C. Rossi, A. Carré, How to choose a suitable NMT solution?: Evaluation of MT quality, in: D. Kenny (Ed.), Machine translation for everyone: Empowering users in the age of artificial intelligence, Language Science Press, Berlin, 2022, pp. 51–79. doi:10.5281/zenodo.6759978.

[45] L. Derczynski, A. Ritter, S. Clark, K. Bontcheva, Twitter part of speech tagging for all: overcoming sparse and noisy data, in: R. Mitkov, G. Angelova, K. Bontcheva (Eds.), Proceedings of the International Conference on Recent Advances in Natural Language Processing, 2013, pp. 198–206.

[46] M. Neunerdt, B. Trevisan, M. Reyer, R. Mathar, Part of speech tagging for social media texts, in: I. Gurevych, C. Biemann, T. Zesch (Eds.), Language Processing and Knowledge in the Web, Lecture Notes in Computer Science 8105, Springer, 2013, pp. 139–150.

[47] T. Gui, Q. Zhang, H. Huang, M. Peng, X. Huang, Part of speech tagging for Twitter with adversarial neural networks, in: M. Palmer, R. Hwa, S. Riedel (Eds.), Proceedings of the 2017 Conference on

Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2017, pp. 2411–2420. doi:10.18653/v1/D17-1256.
[48] Cambridge Dictionary, Cambridge Dictionary, n.d. URL: https://dictionary.cambridge.org/.
[49] WordReference.com, WordReference Dictionary, n.d. URL: https://www.wordreference.com/.