A Comparison of the Word Similarity Measurement in English-Arabic Translation Memory Segment Retrieval including an Inflectional Affix Intervention

Khaled Mamer Ben Milad

[1] Swansea University, UK
882922@swansea.ac.uk

Abstract. The aim of this paper is to investigate the similarity measurement approach of translation memory (TM) in five representative computer-aided translation (CAT) tools when retrieving inflectional verb-variation sentences in Arabic to English translation. In English, inflectional affixes in verbs include suffixes only; unlike English, verbs in Arabic derive voice, mood, tense, number and person through various inflectional affixes e.g. pre or post a verb root. The research question focuses on how the TM matching metrics measure a combination of the inflectional affixes when retrieving a segment. If it is dealt with as a character intervention, are the types of intervention penalized equally or differently? This paper experimentally examines, through a black box testing methodology and a test suite instrument, the penalties that TM systems' current algorithms impose when input segments and retrieved TM sources are exactly the same, except for a difference in an inflectional affix. It would be expected that, if TM systems had some linguistic knowledge, the penalty would be very light, which would be useful to translators, since a high-scoring match would be presented near the top of the list of proposals. However, analysis of TM systems' output shows that inflectional affixes are penalized more heavily than expected, and in different ways. They may be treated as an intervention on the whole word, or as a single character change.

Keywords: Arabic inflectional affix, TM retrieval, TM metrics, penalty imposed

1 Introduction

A translation memory is a database that contains translation units which comprise source language segments aligned with their target language translations. When a new (input) text is uploaded for translating, the TM matching and retrieval mechanism computes the string similarity of the input in comparison with the source segments contained in the TM. Then, TM technology leverages translation candidates with the highest similarity to the input segment [1]. However, there is little detailed information on how the matching algorithms assign a score to the matching strings.

Translation and Interpreting Technology Online, pages 125–141,
July 05-07, 2021.
https://doi.org/10.26615/978-954-452-071-7_014

Most previous studies repeat the belief that the TM similarity measurement is based on the Levenshtein [2] distance algorithm (e.g. Simard and Fujita [3]). This similarity measurement uses three basic edit operations (insertion, deletion and substitution) to determine a distance between two strings, then the distance is normalized into a matching score. Hence, the question is how do TM systems measure the matching between two strings? Is the matching measurement based on a comparison word by word? Or, is the measurement computed character by character? For example, if two source segments are identical except for a difference in an inflectional affix, does the algorithm measure a combination of the inflectional affixes as a word intervention or a character intervention?

The researcher's hypothesis is that the TM metrics may compute inflectional verb variations is either as a word intervention, which means that the algorithm regards the inflected form as a totally different word, where the penalty would be expected to be very heavy, or as a character intervention, in which the penalty would be based on the edit type. Hence, we argue that TM similarity metrics could have difficulties detecting inflectional affixes, which would not result in seeing high-scoring TM proposals.

On the other hand, if the TM system were able to undertake a morphological analysis, it would treat the inflectional affix in a different way. However, Macklovitch and Russell [4] pointed out that one of the limitations of TM systems is the failure to recognize inflectional variants. They argue that despite any necessary minor adjustments, a segment that includes an inflected word is still potentially informative. Somers [5] highlighted that a high-matching technique is needed to use linguistic information such as inflection paradigms, synonyms and grammatical alternations in order to improve TM fuzziness. A fuzzy match means a percentage assigned by a TM metric occurs when the input is partially similar to TM source; if the difference is minor, the value is high. If, on the other hand, the difference is significant, the score is low.

In this paper, we aim to investigate the performance of TM similarity algorithms when retrieving inflectional verb-variation sentences in Arabic-to-English translations. To achieve the aims of the study, a special corpus of Arabic source segments and English target segments is provided, in which we apply a number of inflectional verb-variation transformation rules to the Arabic source segments. Test segments were extracted from the corpus and the edit distance metric was used as an analysis tool.

The paper is organised as follows: Section 2 reviews related works related to semantic matching in TMs. In section 3, we present a review of the verb inflectional affixes in Arabic. We describe the experimental methodology in section 4. We summarise the findings in section 5, and discussion of the results in section 6. Lost usability opportunity of highly similar TM proposals is analysed in section 7. Finally, the conclusions drawn from the research are in section 8.

2 Related studies

Due to the limitation of the TM algorithms, various researchers have focused on how to improve semantic matching in TMs. Gupta et al. [6, 7]; Gupta and Orasan [8] offer a semantically enhanced edit-distance method by introducing a paraphrase data-

base into the edit-distance metric during the matching process. The extra paraphrase TM database contains semantic information such as lexical, phrasal and syntactic paraphrases. Paraphrases in the PPDB dataset are extracted using a statistical method. Both automatic and human evaluation have shown that paraphrasing improves TM matching and retrieval

In very recent research, Ranasinghe et al. [9] claim that most of the methods that try to capture semantic similarity in TM were trialled on small databases and are not appropriate for the large TMs normally employed by translators. These researchers, therefore, have introduced an approach that relies on encoding sentences into embedded vectors in order to improve the matching and retrieval process; this means that text similarity is calculated using deep learning (vector representation) rather than texts. The experiment employed the Universal Sentence Encoder for English released by Google [10]. A test was run on English ↔ Spanish languages pairs, using the DGT-TM of the European Commission's translation service. The results showed that universal sentence encoder architectures handle semantic textual similarity better than the edit distance metrics. The approach is language independence and could be employ to any language pair if there are embeddings available for the source language. It appears to be a promising method for the retrieval of a rich semantic similarity, like Arabic.

Further, Tezcan et al. [11] propose developing a "neural fuzzy repair" method by using sub-word-level segmentation in fuzzy match combination to maximise the coverage of source words. This method employs vector-based sentence similarity metrics for retrieving TM matches in combination with alignment-based features on overall translation quality. This method aims to maximise the added value of retrieved matches within the neural fuzzy repair paradigm. A test was run on eight language combinations: English ↔ Hungarian, English ↔ Dutch, English ↔ French, and English ↔ Polish using the DGT-TM. This study reaffirms the usefulness of fuzzy matching based on vector representations to capture semantic relationships between subwords.

3 Review of Arabic verb inflections

The Arabic language is a highly inflected language, and verb inflection (which is Known in Arabic as الأوزان, al-awzaan) is a conjugation process of creating new stems from the root using specific verbal templates. The verb conjugation involves the creation of new stems from the verb's root (the base of the verb form) using specific verbal templates. Neme [12] explains that the combination of a root with a pattern produces an inflected form in which the root signifies a morphemic abstraction for a verb, while the pattern is a template of characters (indices) surrounding the root consonants.

The verb's tense – and other aspects such as gender and number – are generally represented using the rules of inflectional verb morphemes. Tenses are used in either the perfect or imperfect form; the former indicates the past tense while the latter indicates the present or future tense. The language uses a unique inflection system: for

example, verbs in the past tense are often designated by suffixes, whereas verbs in the present or future tense are often identified by a prefix. Numbers are classified as plural, dual or singular, with two gender categories, feminine and masculine. The number and gender features can be integrated with the verb's tense and expressed in single-word forms [13].

Another important characteristic of Arabic is that the overwhelming majority of verbs have roots consisting of three characters, in which the position of an inflectional affix (i.e. a character) that shapes the template is positioned either as a prefix or a suffix only, while the affix string may encompass one character or more. Habash [14] states in his book 'Introduction to Arabic Natural Language Processing' that verb inflections have a limited number of patterns: ten basic templates for a three-character root and two templates for a four-character root. This means that the triliteral (three-character-root) verb could be transformed from one template into another template just by attaching a prefix (an initial attachment) or a suffix (a final attachment), while the string of basic form stays as one chunk (no mid-form intervention). In Transformation sub-section (4.3) below, we describe a prefix and suffix combination with a three-character root in order to make different verbal templates.

4 Methodology and Experimental Setup

4.1 Evaluation method

The method of TM systems evaluation, which is further illustrated in the subsection on the experimental setup below, was based on the approach of considering the TM as a 'black-box' component advanced by Simard and Fujita [3]

The test segments were extracted from a corpus. The corpus, which was created by the researcher, was imported into the CAT applications as a TM. Then the test segments (i.e. the input) were uploaded as a document to be translated in the selected CAT tool. As a result, the matching scores of TM proposals offered a similarity measurement.

The goal of the study was to initially test then compare the five representative CAT tools in terms of retrieving inflectional verb-variation sentences in Arabic to English translation. Accordingly, the emphasis was on whether the TM could handle the intervention of inflectional affixes in a linguistic analysis or as an edit distance operation.

4.2 Preparing the experimental database

Finding a corpus including specific inflectional verb-variation sentences in Arabic proved difficult; thus, we created our own corpus in order to build more effective and robust results. The size of the corpus was 45 aligned sentences, with Arabic as the source language of the translation units and English as the target, while the segments' length ranged from 3 to 7 words. The procedure of making the Arabic source segment in the corpus was that the verb-stem was generated from a three-character root, combined with a single character as a prefix or suffix. We selected four templates (i.e.,

verb stems) to represent the inflectional verb variations. At least three samples were used in each event. We are aware that the corpus created was very small, therefore, we regard this work and the results as preliminary.

4.3 Transformation

For the purpose of the study, the four templates selected were transformed from perfective to imperfective or vice versa by changing their inflectional affix. The change of character led to a change in the verb tense only, while the aspects of the subject remained the same. We explain below the rules of transformation by using the canonical verb (فعل), (do), which is commonly used by Arabic grammarians in creating verb templates:

- Rule 1: The verb template (VT) of the source segment was changed from an imperfective (third person masculine) into a perfective pattern: يفعل (He does)> فعل or فعل (He did). The transformation was made by dropping an initial character ي (a single character prefix), or sometimes by adding a diacritic mark on the final-character ـَل. However, the insertion of a diacritic mark is optional in Arabic, and it may be omitted from the text. For example, 'يشرب الطفل الحليب الطازج صباحا.' / yashrab altifl alhalib altaazij subahana / 'The child drinks fresh milk in the morning'. In such example, if the prefix (يـ) is removed (deletion operation), the tense of the sentence changes into past 'يشرب الطفل الحليب الطازج صباحا ا.'[1] / shrab altifl alhalib altaazij subahana / 'The child drank fresh milk in the morning'. In the experiment, we removed such prefixes, so that the input string was different from the TM source by a single character. Table 1 below shows the verb template transformation process.
- Rule 2: In contrast to Rule 1, the verb template was changed from a perfective (third person masculine) into an imperfective pattern, فعل or فعلَ (He did)> يفعل (He does), by adding an initial-character ي (a single-character prefix). Table 1 below shows the verb template transformation process.
- Rule 3: The verb template of the source segment was changed from a perfective (third person feminine) into an imperfective pattern, فعلت (She did) > تفعل (She does), by changing a final character ت (a single-character suffix) into an initial character ت (a single-character prefix). Table 1 below shows the verb template transformation process.
- Rule 4: In contrast to Rule 3, the verb template was changed from an imperfective (third person feminine) into a perfective pattern: تفعل (She does) > فعلت (She did). The change was made by changing an initial character ت a (single-character prefix) into a final character ت (a single-character suffix). Table 1 below shows the verb template transformation process.

[1]Track Changes was used for the intervention.

Using an Arabic verb conjugator website,[2] the automated ACON application can conjugate the different templates of the Arabic verb by selecting the root and the type. Table 1 below shows the transformation of four templates in Arabic sentences using edit operations.

Table 1. Transformation of four verb templates in Arabic sentences using edit operations

Rule	Original VT	Morphological intervention	Edit distance	Transformed VT
1	يفعل [He does]	Dropping prefix	Deletion	فعل or فعلَ[3] [He did]
2	فعل or فعلَ[4] [He did]	Adding prefix	Insertion	يفعل [He does]
3	فعلت [She did]	Shifting suffix into prefix	Substitution	تفعل [She does]
4	تفعل [She does]	Shifting prefix into suffix	Substitution	فعلت [She did]

After applying the rules listed above each sentence of the test underwent a transformation, which converted linguistically the imperfective pattern of the verbs in the original sentences into the perfective patterns or vice versa using one type of edit operation. Then, the modified test segments, which were used as a document to be translated, were run against the TM corpus which included the original segments.

The verb templates in Table 1 above that represent the verb inflections in Arabic have the structure of the research query; the transformation of verb templates represents the rich morphology of the language; the edit operations potentially represent the similarity measurement used by translation memory systems.

4.4 Experiment with pre-translation

Having processed the test segments, they were then submitted to the CAT applications as files to be translated. If we had to translate again a segment from the source language, the match would obviously be 100%. The translation project in each CAT application was based on the corpus created as a TM file that included the original segments; to make the comparison as fair as possible, the same input text (test segments) was uploaded as a file for translation in the five CAT tools. Then, a pre-translation was processed to gain the TM matching scores.

[2]ACON, the Arabic Conjugator - conjugate Arabic verbs online (baykal.be)
[3]The diacritic mark of *fatha*
[4]The diacritic mark of *fatha*

The input text, which contained 45 segments, was translated by five CAT tools: Déjà Vu X3 (hereafter referred to as DVX3);[5] OmegaT;[6] memoQ 9.0;[7] Memsource Cloud;[8]and Trados Studio 2019.[9] These CAT tools, widely used by professional translators [15], produced fuzzy matches that were analysed according to their results. As the test segments and TM source were identical except for a difference in an inflectional affix, it was desirable for the TM similarity metrics to produce a very high score which could be presented at the top of the list of proposals presented to the translator.

5 Findings

This section displays the results obtained from the TM systems' attempts to retrieve matches for the test segments. We assumed that scores at the higher end are better, for example 95% is better than 80%.

5.1 Déjà Vu X3 Scoring

The matches retrieved by DVXwere found to occupy a consistent band according to the length of the test segments and whether they contained an inflectional affix intervention (deletion, insertion, or substitution). The matching scores decreased in a consistent way as the number of words in the segment decreased and ranged from 67% to 86%. Figure 1 (below) illustrates the fuzzy matching scores (three samples were used in each event) that each segment length (SL) supplied due to their inflectional affix combination (inserting a one-character prefix, deleting a one character prefix, and shifting one character into suffix or vice versa).

The figure below clearly shows that DVXtreated the test segments equally regardless of the type of inflectional affixes intervention. Further, the retrieved matches of three-to-seven-word segments were distributed among the different fuzzy bands. For example, 67% provided a low fuzzy score (i.e. a 33% penalty per one-edit operation), while for seven-word segments, 86% provided a high fuzzy score (i.e. a 16% penalty per one-edit operation, or approximately one word in seven). This means that TM users may not see proposals of high fuzzy matches for short sentences that have just a single character difference.

5.2 memoQ 9.0 scoring

The scores of memoQ were categorised in two phases. The matching scores of memoQ were derived from two different ranges: a low match range and a high match

[5]https://atril.com/
[6]https://omegat.org/
[7]https://www.memoq.com/memoq-versions/memoq-9-5
[8]https://www.memsource.com/
[9]https://www.trados.com/products/trados-studio/

range. The five-, six- and seven-word segment routines were in the low fuzzy range, while the three- and four-word segments were given a relatively high fuzzy range whether these segments contained an inflectional affix intervention (deletion, insertion, or substitution). The match scores ranged from 77% to 91%. Figure2 (below) illustrates the different range of matches for each segment length (SL) provided.

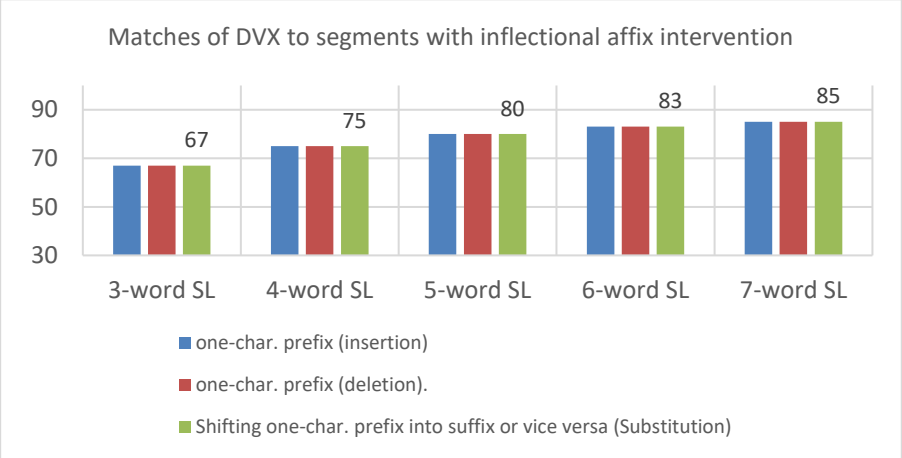

Fig.1. DVX matching scores for 3-to-7-word segment lengths (SL) with an inflectional affix intervention.

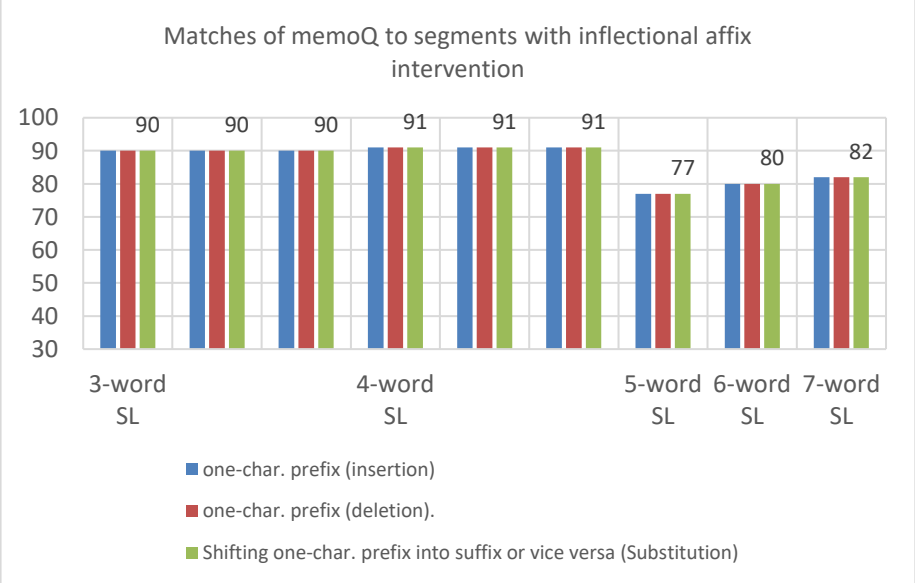

Fig. 2. memoQ matching scores for 3-to-7-word segment lengths (SL) with an inflectional affix intervention

As the figure above shows, the matches of three- and four-word segments with an inflectional affix were retrieved in a high fuzzy match. For example, the three-word and four-word segments were provided with a 90% and 91% match, respectively (i.e., a 10% and 9% penalty). In terms of the segments of five words and above, the scores unexpectedly matched lower regardless of the edit operation. For example, five-word segments provided a match of 77% (i.e., a 23% penalty).

This suggests that the retrieval of segments of five words or above was based on the number of words, while the retrieval of three- and four-word segments was not. It seems that the measurement was based on the total number of characters. This may explain the difference in the matching levels: the character-based measurement produced considerably better results. As a result, the short segments would be offered in a high fuzzy band, while longer segments would be scored lower, although in all cases the difference was just a single character.

5.3 Memsource Cloud scoring

The TM system of Memsource retrieved the test segments in an inconsistent range of scores. Thus, the experiment used the filter feature in the system's setting to sort the source's shortest segment first, which was based on the number of characters. When observing the fuzzy matches, the scores appeared to decrease as the total number of characters in the segment fell regardless of how many words a segment contained. Similarly, when the source was sorted according to the principle of the longest first, the matches appeared to increase as the total number of characters in the segment increased. As A result, the matches appeared to rely in the first place on the total number of segment characters, and in the second place on the position of the edit operation. Further, the match values decreased as the total number of characters decreased; the length of segments varied from 16 to 49 characters (i.e., both characters and whitespaces), while the match scores varied between 73% and 98%. Due to these scattered scores, the matches illustrated in Figure 3 are presented as a chart, using a line with markers: the markers represent the inconsistency of scores, while the lines represent the impact of the segment length.

As Figure 3 shows, it is obvious that the retrieval of segments with a one-character prefix were given high percentages, whereas the operation of shifting a one-character prefix into a suffix position, or vice versa, was assigned a lower fuzzy band.

For example, the matches of segments ranging from 49 to 16 characters, produced by inserting a one-character prefix, ranged from 98% to 94%, whereas segments ranging from 49 to 76 characters. produced by deleting a one-character prefix, also scored between 98% and 94%. Shifting a one-character prefix into a suffix position, or vice versa, produced match scores in the lower fuzzy band. For instance, segments ranging from 46 characters to 18 characters produced scores between 90% and 73% when a one-character prefix was changed into a suffix, whereas segments ranging from 46 characters to 19 characters produced scores between 91% and 74% when a one-character suffix was changed into a prefix.

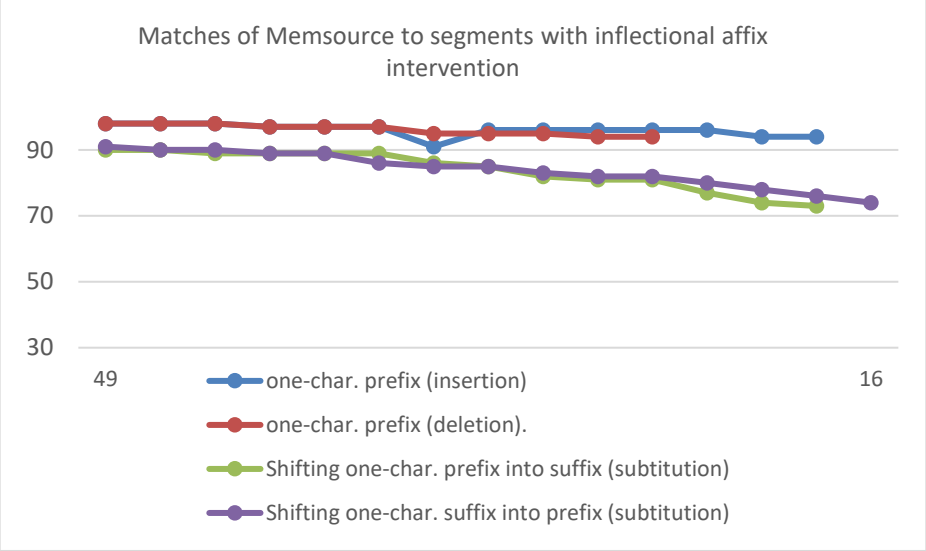

Fig. 3. Memsource matching scores for a segment 49-16 characters long due to changes to an inflectional affix.

The explanatory hypothesis is that, on the one hand, a one-character prefix was dealt with as a one-edit operation, while changing a one-character prefix into a suffix, or vice versa, was treated as a two-edit-operation. On the other hand, editing a one-character prefix occurred on the word-initial position, while changing a one-character prefix into a suffix, or vice versa, occurred on the word-initial and word-final positions. This suggests that the matching metrics dealt with the impact of a prefix combination in a different way to that of a suffix combination. As a result, the retrieval of segments with an inflection affix would be offered at a high fuzzy level under specific conditions. However, further research is needed to confirm this hypothesis since this study is based on the number of words in segments.

5.4 OmegaT scoring

The fuzzy matches provided by OmegaT were relatively high; however, they dropped gradually as the segment became shorter, whether it contained a deletion, insertion or substitution operation. The matching scores consistently related to the segments' word length – the scores ranged from 83% to 92%. Figure4 (below) shows the matching values for each segment length (SL) according to the editing of an inflectional affix.

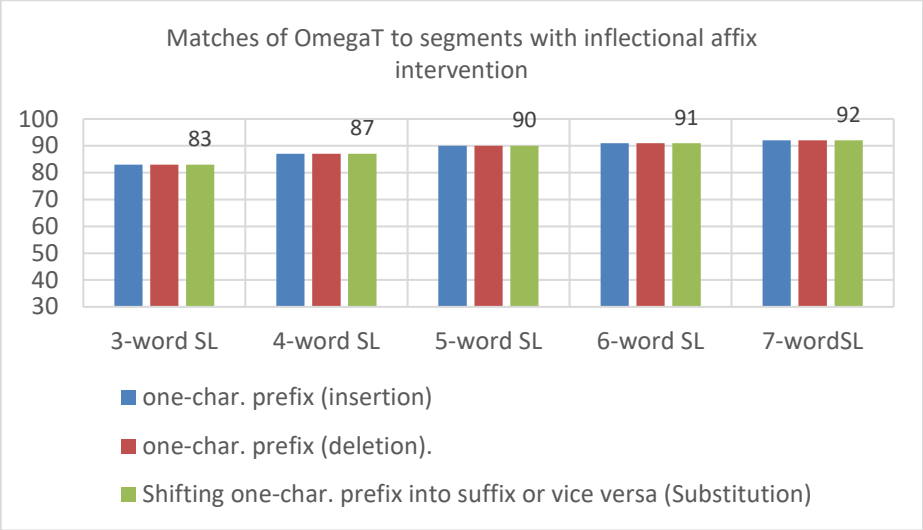

Fig. 4. OmegaT matching scores for 3-to-7-word segment lengths (SL) with an inflectional affix intervention.

As Figure 4 clearly shows, OmegaT's matching metrics dealt with the different ways of editing the inflectional affix in the same fashion, retrieving four- to seven-word segments in a high fuzzy band; only the three-word routine was placed in the middle fuzzy band. This means that OmegaT would retrieve segments with an inflectional affix – except for a three-word routine – in a high fuzzy band, which would be very useful from the perspective of translators.

5.5 Trados Studio 2019 scoring

The matching scores produced by Trados Studio also fell steadily as the segment length became shorter, whether these segments contained a deletion, insertion or substitution operation. The matching values were consistently related to the segment's word length. The match scores ranged from a 78% to 91%. Figure 5 (below) displays the matching values for the retrieval for each segment length (SL).

It can be seen that Trados Studio dealt with the retrieval of segments with an inflectional affix in the same way regardless of the type of character-edit operation involved. The matches were distributed between middle and high fuzzy bands, where the three- and four-word segments matched 78% and 83%, respectively (i.e. in the middle fuzzy band), and the five- six- and seven-word segments scored in a high fuzzy band. This means that TM users would not see three- and four-word segments with only a one-character difference in the high fuzzy band range.

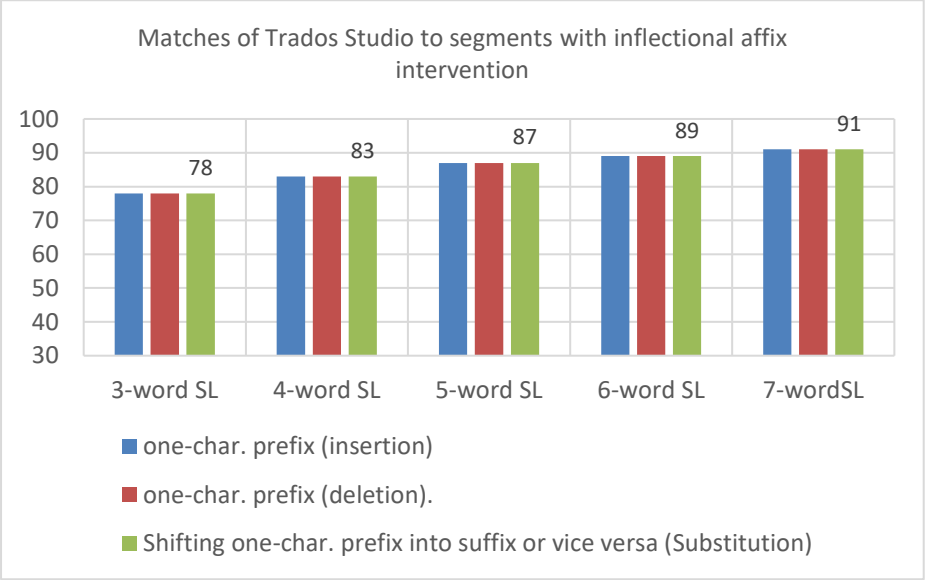

Fig. 5. Trados Studio matching scores for 3-to-7-word segment lengths (SL) with an inflectional affix intervention.

The results showed that the various TM systems differed in their handling of diacritic marks. First, the algorithm of DVX, OmegaT, Trados Studio systems and the scoring of five- to seven-word segments in memoQ, which produced consistent matches according to the segments' word length, did not appear to be influenced by the insertion or removal of diacritic marks – the matches retrieved were the same. Secondly, the metrics of Memsource and the scoring of three- or four-word segments in memoQ, whose character-based algorithm provided inconsistent values, were affected by a combination of diacritic markers. When calculating segments with and without a diacritic mark using a Levenshtein website,[10] the URL estimated a diacritic marker as a one-edit distance. Hence, a diacritic mark was treated as equal in weight to a one-character intervention in character-based metrics.

6 Discussion

The experiment's findings show that the TM systems treated a combination of inflectional affixes in different ways: the TM matching algorithms dealt with the morphological combination as an intervention on the whole word, as a single character change, or according to the position of the intervention. In all the systems, however, it appears that segment length had a bearing on the results.

[10] https://planetcalc.com/1721/

These findings prompted a comparative analysis of each TM's retrieval of fuzzy bands. This was accomplished by using the length of each segment and the affix position and type as independent variables.

Turning to the DVX results first, it seems that the TM system's algorithm dealt with the inflectional affix as an intervention on the whole word. To account for this, a procedure calculating the surface form of the strings was used. In five-word segments, for example, DVX provided an 80% match (i.e., a 20% penalty). This may be explained by the fact that the algorithm estimated that a four-word string was identical to a five-word string, while a one-word string was non-similar (i.e.$\frac{\mathbf{4}}{\boldsymbol{SL}\,\mathbf{5}} = \frac{\mathbf{80\%}}{\mathbf{100}}\ \boldsymbol{identical\ vs.}\ \frac{\mathbf{1}}{\boldsymbol{SL}\,\mathbf{5}} = \frac{\mathbf{20\%}}{\mathbf{100}}$ non-similar). This implies that the DVX metrics recorded the edit operation (i.e., the inflectional affix) as an intervention on the whole word, resulting in low scores for segments that have a small number of words and an increase in scoring for longer segments.

The reason behind the OmegaT and Trados Studio results could be that their TM similarity algorithms are not only based on the number of words but also employ a specific mechanism for an individual edit operation (i.e., a single-character intervention) to measure the segments' similarity. In five-word segments, for example, any type of character editing (i.e., insertion, deletion, or substitution) was penalised 10% and 13% in OmegaT and Trados Studio, respectively; however, the matching scores provided were consistently in line with the segment's word length whatever the number of characters, which resulted in decreasing scores for short segments and increasing scores for longer ones. However, a comparison of the matching mechanisms of the two systems shows that OmegaT outperformed Trados Studio; the lowest match was scored 83% by OmegaT and 78% by Trados Studio, whereas the highest scores were 92% and 91% for OmegaT and Trados Studio, respectively.

As for the scores of memoQ, in terms of consistent scores, the system algorithm seems to use g an internal mechanism to compute a combination of inflectional affixes in segments of five words or above. The mechanism produced the lowest average scores for the five-, six- and seven-word routines compared with the other systems that provided consistent scores. With a five-word routine, for example, memoQ supplied a 77% match (a 23% penalty) whatever the type of character editing. The penalties imposed by DVX, Trados Studio and OmegaT were 20%, 13% and 10%, respectively. The penalty imposed by memoQ was the heaviest. This means that the similarity algorithms in memoQ, where the measurement was word-based, imposed the heaviest penalty due to the character combination. In terms of the inconsistent matches (i.e., the three- and four-word segments), the matches were retrieved with high percentages despite the short segment length. This may be explained by the fact that the recall was based on the number of characters.

Memsource's matches, which were apparently inconsistently produced according to the number of characters, showed that the retrieval of segments with the insertion or removal of a one-character prefix gave high percentage scores, while the operation of substituting one character produced a lower percentage. It seems that Memsource's retrieval mechanism penalised a prefix combination relatively lightly. This was calculated not according to a linguistic analysis but from the perspective that a prefix combination may cause less damage to the word form than a suffix combination. As a

result, in some cases, the TM matching measurement performed well when a one-character prefix (i.e., inflectional affix) was inserted or removed, but not a one-character suffix.

Overall, the different tools appear to have different routines for handling such inflectional affix interventions. Although none of them is fully satisfactory, especially for short segments, Memsource outperformed the other systems when the intervention of an inflectional affix was a prefix only. The metrics of memoQ penalised the heaviest when the system provided consistent matches. In all the TM systems, the matching scores reduced as the length of the segments decreased but it was seen most clearly in the systems that produced consistent matches. To bear in mind, the study used a very short root – a three-character word including a single character combination, the retrieval of a longer base-form including a prefix or suffix combination may be scored differently by TM systems' algorithms.

To summarise, the TM matching measurements failed to recognise inflectional affixes. This outcome is in line with the results of the studies conducted by Macklovitch and Russell [4] and Planas and Furuse [16], which found that one of the limitations of TM systems is their inability to recognise inflectional variants when retrieving stored data. The current study has provided further experimental evidence, gathered from the scores supplied by five CAT applications, showing that TM matching metrics are not good at distinguishing morphological combinations.

7 Lost usability opportunity

From a usability point of view, the test results show that, although the translator would potentially spend less time and effort editing the inflectional verb-variation segments, they could miss out on seeing those TM proposals because of their low scores. What the users of TM would expect – from a translator's perspective – is that TM algorithms would retrieve inflectional verb-variation segments with a very high match score (i.e., a range of high fuzzy or 85%-94%) since these would need only one edit operation to be identical to the input text. The impact of high fuzzy matches appears in the translation cost. Contrary to this expectation, however, it appears that a translator working with short segments will not be shown a high but a low fuzzy proposal, which may result in the proposals being lost. Hence, the project manager, when preparing a report, may produce inappropriate fuzziness percentages for the translation of a text with a rich morphology including segments with inflectional verb variations, and the price they quote for the translation will consequently be higher than it should be. Table 2 shows the bands of fuzzy matches, according to Studio Trados,[11]produced for the test segments reported by each TM system.

[11]Fuzzy match grids in SDL Trados Studio | Signs & Symptoms of Translation (signsandsymptomsoftranslation.com)

Table 2. Fuzzy match bands as computed by each TM system

Fuzzy bands	**Range of scores**	**DVX**	**memoQ**	**Mem-source**	**OmegaT**	**Trados Studio**
Nearly exact match	95% - 99	0	0	20	0	0
High fuzzy band	85% - 94	12	24	26	48	36
Middle fuzzy band	75% - 84	36	36	10	12	24
Low fuzzy band	50% - 74	12	0	4	0	0
No match	0 - 49%	0	0	0	0	0
Total	Total	60	60	60	60	60

Table 2 displays the ways in which the TM systems differed in fuzzy-match distribution. OmegaT showed a significantly higher number of matches for the high fuzzy band (85-99%), followed by Memsource, while DVX ended up with a significantly smaller number than the other bands. The fuzzy matches varied in distribution according to the different TM systems:

- OmegaT retrieved only 12 out 60 segments, representing 20%, in a lower fuzzy band. These results appear to be the best.
- Memsource retrieved 14 out of 60 segments, representing 24%, in a lower fuzzy band; however, the high fuzzy scores were mainly produced when the intervention was a prefix.
- Trados Studio retrieved 24 out of 60 segments, representing 40%, in a lower fuzzy band.
- memoQ retrieved 36 out of 60 segments, representing 60%, in a lower fuzzy band.
- DVX retrieved 48 out of 60 segments, representing 80%, in a lower fuzzy band. These results are the worst.

As mentioned above, because the fuzzy match levels play a significant role in the calculation of translation costs, these results would have a definite impact on the discount applied to texts that are rich in morphological combinations. Preventing segments that include an inflection affix from ranking as a high fuzzy match would therefore impact the efficiency, consistency and cost of a translation.

8 Conclusion

The overall conclusion drawn from the results of testing the retrieval of TM sources for a text that is rich in morphological combinations is that all the selected

systems revealed a deficiency when it came to identifying inflectional affixes, although OmegaT and Memsource returned more than three-quarters of segments in the high fuzzy band, and memoQ produced considerably better scores to short segments than longer segments. The overall matching scores appeared to be based purely on the string of surface forms and the internal machinery of each system's algorithm, without any linguistic analysis. Hence, the findings substantiate the proposals that implementation of deep learning and vector representations would help capture semantic textual similarity for TM matching. The outcome shows that an inflectional affix intervention was treated as either an intervention on a whole word or a single character change. Consequently, the high matching of retrieved inflectional verb-variation segments in an Arabic-to-English translation would depend on the segment length and the position of the intervention. Further work is needed to extend the investigation to other morphologically rich languages, different positional affixes and longer string formations such as a noun derivation. The findings substantiate the proposals that implementing of encoding sentences into embedded vector should be incorporated into similarity metrics of TM systems.

Acknowledgments

This work is a part of the Ph.D. research at Swansea University, UK. The author would like to extend special thanks to supervisors Prof. Andrew Rothwell and Dr. Maria Fernandez Parra for their helpful comments on the content of this paper. The PhD research programme is funded by High Education Ministry, Libyan Government.

References

1. Vázquez, L. M.: An empirical study on the influence of translation suggestions' provenance metadata. (2012).
2. Levenshtein, Vladimir I.: Binary codes capable of correcting deletions, insertions, and reversals. In Soviet physics doklady, vol. 10, no. 8, pp. 707-710. (1966)
3. Simard, Michel, and Fujita, A.: A poor man's translation memory using machine translation evaluation metrics. Proceedings of the 10th Conference of the Association for Machine Translation in the Americas: Research Papers. (2012)
4. Macklovitch, E., and Russell, G: What's been forgotten in translation memory. Conference of the Association for Machine Translation in the Americas. Springer, Berlin, Heidelberg, (2000)
5. Somers, H.: Translation memory systems. Benjamins Translation Library 35. (pp.31-48) (2003):
6. Gupta, R, Orăsan, C., Zampieri, M., Vela, M., and Van Genabith, J.: Can Translation Memories afford not to use paraphrasing?. In Proceedings of the 18th Annual Conference of the European Association for Machine Translation. (2015)
7. Gupta, Rohit, Orăsan, C., Liu, Q., and Mitkov, R.: A Dynamic Programming Approach to Improving Translation Memory Matching and Retrieval Using Paraphrases. In International Conference on Text, Speech, and Dialogue (pp. 259-269) Springer, Cham (2016)

8. Gupta, R., Orăsan, C.: Incorporating Paraphrasing in Translation Memory Matching and Retrieval. In: Proceedings of the 17th Annual Conference of the European Association for Machine Translation (pp. 3–10) EAMT-2014. Dubrovnik, Croatia: European Association for Machine Translation. (2014)
9. Ranasinghe, Tharindu, Orăsan, C., and Mitkov, R.: Intelligent Translation Memory Matching and Retrieval with Sentence Encoders. arXiv preprint arXiv:2004.12894 (2020).
10. Cer, Daniel, Yang, Y., Kong, S., Hua, N., Limtiaco, N., St John, R., Constant, N, Guajardo-Cespedes, M., Yuan, S., Tar, C. and Strope, B: Universal sentence encoder for English. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations (pp. 169-174) (2018)
11. Tezcan, A., Bulté, B. and Vanroy, B.: Towards a Better Integration of Fuzzy Matches in Neural Machine Translation through Data Augmentation." In Informatics, vol. 8, no. 1, p. 7. Multidisciplinary Digital Publishing Institute (2021)
12. Neme, A. A.: A lexicon of Arabic verbs constructed on the basis of Semitic taxonomy and using finite-state transducers. In WoLeR 2011 at ESSLLI International Workshop on Lexical Resources. (2011)
13. Habash, N. and Rambow, O.: Morphophonemic and orthographic rules in a multi-dialectal morphological analyzer and generator for arabic verbs. In International symposium on computer and arabic language (iscal), riyadh, saudi arabia. (2007)
14. Habash, N. Y.: Introduction to Arabic natural language processing. Synthesis Lectures on Human Language Technologies 3, no. 1 (pp. 1-187) (2010)
15. Moorkens, J. and O'Brien, S.: Assessing user interface needs of post-editors of machine translation." In Human issues in translation technology (pp. 127-148) Routledge (2017)
16. Planas, E. and Furuse, O.: Formalizing translation memories. In Machine Translation Summit VII, vol. 1999 (pp. 331-339) (1999)