ISSN 2395-8618

# Towards Recognizing Phrase Translation Processes: Experiments on English-French

Yuming Zhai, Pooyan Safari, Gabriel Illouz, Alexandre Allauzen, and Anne Vilnat

***Abstract*—When translating phrases (words or group of words), human translators, consciously or not, resort to different translation processes apart from the literal translation, such as Idiom Equivalence, Generalization, Particularization, Semantic Modulation, *etc.* Translators and linguists (such as Vinay and Darbelnet, Newmark, *etc.*) have proposed several typologies to characterize the different translation processes. However, to the best of our knowledge, there has not been effort to automatically classify these fine-grained translation processes. Recently, an English-French parallel corpus of TED Talks has been manually annotated with translation process categories, along with established annotation guidelines. Based on these annotated examples, we propose an automatic classification of translation processes at subsentential level. Experimental results show that we can distinguish non-literal translation from literal translation with an accuracy of 87.09%, and 55.20% for classifying among five non-literal translation processes. This work demonstrates that it is possible to automatically classify translation processes. Even with a small amount of annotated examples, our experiments show the directions that we can follow in future work. One of our long term objectives is leveraging this automatic classification to better control paraphrase extraction from bilingual parallel corpora.**

***Index Terms*—Translation processes, non-literal translation, automatic classification.**

## I. Introduction

SINCE 1958, translators and linguists have published work on translation processes [1], [2], [3], [4]. They distinguish literal translations from other translation processes at subsentential level. Consider these two human non-literal translation examples: the first translation preserves exactly the meaning, where the fixed expression *à la hauteur de* 'to the height of' has a figurative sense which means *capable of solving*; while the second one is more complicated, there exists a textual inference between the source text and the translation.

*(1.EN) a solution* ***that's big enough to solve*** *our problems*
*(1.FR) une solution* ***à la hauteur de*** *nos problèmes*
*(2.EN) and* ***that scar has stayed with him*** *for his entire life*
*(2.FR) et que, toute sa vie,* ***il a souffert de ce traumatisme***
('he has suffered from this traumatism')

Non-literal translations can bring difficulties for automatic word alignment [5], [6], or cause meaning changes in certain cases. However, to the best of our knowledge, there has not been effort to automatically classify these fine-grained translation processes to benefit downstream natural language processing tasks. For example, Machine Translation (MT) techniques have been leveraged for paraphrase extraction from bilingual parallel corpora [7], [8]. The assumption is that two monolingual segments are potential paraphrases if they share common translations in another language. Currently the largest paraphrase resource, PPDB (ParaPhrase DataBase) [9], has been built following this method. Nonetheless, Pavlick *et al.* [10] revealed that there exist other relations (*i.e. Entailment (in two directions), Exclusion, Other related and Independent*)[1] than strict equivalence (paraphrase) in PPDB. Non-literal pivot translations inside the parallel corpora could break the strict equivalence between the candidate paraphrases extracted, whereas they have not received enough attention during this corpora exploitation.

From a linguistic point of view, apart from the word-for-word literal translation, different versions of human translations reflect the richness of human language expressions, where various translation processes could be employed. Furthermore, because of the existing differences between languages and cultures, non-literal translation processes are sometimes inevitable to produce correct and natural translations. The fine-grained phrase-level translation processes could help foreign language learners to better compare the language being learned with another language already mastered.

Based on the theories developed in translation studies and through manually annotating and analyzing an English-French parallel corpus, Zhai *et al.* [11] have proposed a typology of translation processes adapted to their corpus. In this work our main contribution is proposing an automatic classification of translation processes at subsentential level, based on these annotated examples. From the aspect of granularity and our goal of better controlling paraphrasing process or helping foreign language learners, it is different from the task of filtering semantically divergent parallel sentence pairs to improve the performance of MT systems [12], [13], [14]. Experimental results show that we can distinguish non-literal translation processes from literal translation with an accuracy of 87.09%, and 55.20% for classifying among non-literal multi-classes.

Manuscript received on June 10, 2019, accepted for publication on September 7, 2019, published on December 30, 2019.

The authors are with LIMSI-CNRS, Univ. Paris-Sud, Univ. Paris-Saclay, France (e-mail: {firstname.lastname}@limsi.fr).

The first two authors contributed equally to this article.

[1]Exclusion: X is the contrary of Y; X is mutually exclusive with Y. Other related: X is related in some other way to Y. (*e.g. country / patriotic*). Independent: X is not related to Y.

ISSN 2395-8618

In the present paper, after reviewing related work, we describe the manual annotation and the data set. Exploited features and different neural network architectures will be presented, followed by experimental results and error analysis. Finally we conclude and present the perspectives of this work.

## II. Related Work

Translators and linguists have proposed several typologies to characterize different translation processes. Vinay and Darbelnet [1] identified direct and oblique translation processes, the latter being employed when a literal translation is unacceptable, or when structural or conceptual asymmetries arising between the source language and the target language are non-negligible. Following studies include, among others, the work of Newmark [2], [15], Chuquet and Paillard [3]. More recently, Molina and Hurtado Albir [4] proposed their own categorization based on studying the translation of cultural elements in the novel *A Hundred Years of Solitude* from Spanish to Arabic.

Non-literal translations or cross-language divergences have been studied to improve MT related techniques. In order to enable more accurate word-level alignment, Dorr *et al.* [5] proposed to transform English sentence structure to more closely resemble another language. A translation literalness measure was proposed to select appropriate sentences or phrases for automatically constructing MT knowledge [16]. Using a hierarchically aligned parallel treebank, Deng and Xue [6] semi-automatically identify, categorize and quantify seven types of translation divergences between Chinese and English.[2] Based on the syntactic and semantic similarity between bilingual sentences, Carl and Schaeffer [17] developed a metric of translation literality. We have drawn inspiration from these preceding work for our feature engineering.

Recently, different models have been proposed to automatically detect translation divergence in parallel corpora, with the goal of automatically filtering out divergent sentence pairs to improve MT systems' performance. An SVM-based cross-lingual divergence detector was introduced [12], using word alignments and sentence length features. Their following work [13] proposed a Deep Neural Network-based approach. This system could be trained for any parallel corpus without any manual annotation. They confirmed that these divergences are a source of performance degradation in neural machine translation. Pham *et al.* [14] built cross-lingual sentence embeddings according to the word similarity with a neural architecture in an unsupervised way. They measure the semantic equivalence of a sentence pair to decide whether to filter it out.

Another task studying human translations concerns automatic post-editing [18]. The aim is evaluating systems for automatically correcting translation errors of an unknown "black box" MT engine, by learning from human revisions of translations produced by the same engine. Evaluation metrics include TER [19], BLEU [20] and manual evaluation. The task that we propose here is different from these attempts, which either filter semantically divergent sentence pairs to improve the performance of MT systems; or automatically correct machine translation errors to improve the translation quality. Our task of classifying translation processes (in two classes or in multi-classes) at subsentential level is a stand-alone task. One of our long term objectives is leveraging this automatic classification to better control phrase-level paraphrase extraction from bilingual parallel corpora.

## III. Manual Annotation and Data Description

In order to model translation choices made by human translators at subsentential level, Zhai *et al.* [11] have annotated a trilingual parallel (English-French, English-Chinese) corpus of TED Talks[3] with translation processes. The corpus is composed of transcriptions and human translations of oral presentations. The inter-annotator agreement (Cohen's Kappa) [21] for annotating the English-French and English-Chinese control corpus is 0.67 and 0.61, both around the substantial agreement threshold. This indicates that the task of manual annotation is already complicated. Readers can find more details of corpus construction in the article [11].

The automatic classification is conducted on the English-French pair in this work. We present in the table I a brief definition, a typical example and the number of instances for each category to be automatically classified.[4] We combine *Transposition* and *Mod+Trans* in a category *Contain_Transposition*, where *Modulation* is considered as a neutral part. We will work on the classification of the pair English-Chinese once the annotation phase is finished. In this work, we conduct experiments in a simplified scenario, where we already know the boundaries of bilingual pairs, and we only predict the translation process. For example, given the pair *deceptive* $\rightarrow$ *une illusion* in a pair of bilingual sentences, the goal is to predict its label *Contain_Transposition*.

## IV. Automatic Classification

We have tried two approaches for the automatic classification. Since the size of the cross validation data set is quite small, we first compare different statistical machine learning techniques with feature engineering. We also build different neural network architectures which we explain below.

### A. Feature Engineering with Statistical Machine Learning Techniques

We describe below the features exploited in this work. The tag sets of English and French for part-of-speech (PoS)

[2]Lexical encoding; difference in transitivity; absence of language-specific function words; difference in phrase types; difference in word order; dropped elements; structural paraphrases.

[3]https://www.ted.com/

[4]Note that there are other detailed annotation rules in the annotation guidelines.

ISSN 2395-8618

TABLE I
DEFINITION, TYPICAL EXAMPLE AND NUMBER OF INSTANCES FOR EACH TRANSLATION PROCESS TO BE AUTOMATICALLY CLASSIFIED. THE INSTANCES WERE MANUALLY ANNOTATED IN AN ENGLISH-FRENCH PARALLEL CORPUS OF TED TALKS. WE COMBINE *Transposition* AND *Mod+Trans* IN A CATEGORY *Contain_Transposition* FOR THE AUTOMATIC CLASSIFICATION.

| Translation Process | Definition and typical example |
|---|---|
| Literal (3771) | Word-for-word translation, also concerns lexical units in multiword form. *certain* ***kinds*** *of* → *certains* ***types*** *de* |
| Equivalence (289) | Non-literal translation of proverbs or fixed expressions; a word-for-word translation makes sense but the translator expresses differently, without changing the meaning and the grammatical classes. ***back then*** → ***à l'époque*** *('at that time')* |
| Generalization (86) | Several source words or expressions could be translated into a more general target word or expression, the translator uses the latter to translate. *as we* ***sit here*** *in ...* → *alors que nous* ***sommes*** *à ... ('as we are at ...')* |
| Particularization (215) | The source word or expression could be translated into several target words or expressions with a more specific meaning, and the translator chooses one of them according to the context. *the idea I want to* ***put out*** *is ...* → *l'idée que je veux* ***diffuser*** *c'est ... ('the idea I want to spread is ...')* |
| Modulation (195) | Metonymical and grammatical modulation [3]; change the point of view; the meaning could be changed. ***that scar has stayed with him*** → ***il a souffert de ce traumatisme*** *('he has suffered from this traumatism')* |
| Transposition (289) | Change grammatical classes without changing the meaning. *unless* ***something changes*** → *à moins qu'****un changement ait lieu*** *('unless a change occurs')* |
| Mod+Trans (53) | Combine the transformations of *Modulation* and of *Transposition*, which could make the alignment difficult. *this is a completely* ***unsustainable*** *pattern* → *il est absolument* ***impossible de continuer sur*** *cette tendance ('it is completely impossible to continue on this trend')* |

tagging, constituency parsing and dependency parsing have been converted into three compact and unified tag sets [22].

1) The PoS tagging is done by *Stanford CoreNLP* [23] for the two languages. On source and target side, for each PoS tag, the number of its occurrence is counted in a vector. We also calculate the cosine similarity between these two vectors (on all words and only on content words).[5]

2) We verify the pattern of PoS tag sequence changing according to a manual list, for example the pair *methodologically* → *de façon méthodologique* 'methodologically' corresponds to the pattern *ADV* → *ADP NOUN ADJ*.

3) The number of tokens in the two segments ($l_e$, $l_f$), the ratio of these numbers ($l_e/l_f$, $l_f/l_e$), the distance Levenshtein [24] between the segments.

4) The constituency parsing is done by *Bonsai* [25] for French, by *Stanford CoreNLP* for English. We compare the PoS tags for a pair of words, the non-terminal node tags for a pair of segments, the tag category (*e.g.* verb → verb phrase) for a word translated by a segment or vice versa.

5) The dependency parsing is done by *Stanford CoreNLP* for the two languages. Inside the segments, the number of occurrence of each dependency relation is counted. Outside the segments, among the words linked at source and target side, we filter those which are aligned in the sentence context. Then the number of occurrence of each dependency relation between the words in segments and these context words is counted.

6) The cosine similarity is calculated between the embeddings from *ConceptNet Numberbatch* [26]. This resource is multilingual and the system based on *ConceptNet* took the first place in the task "Multilingual and Cross-lingual Semantic Word Similarity" of SemEval2017 [27], [28]. Certain multi-word expressions have their own embeddings in this resource. Otherwise, we calculate the average of embeddings only on content words. The same features are calculated for lemmatized segments.[6]

7) The resource *ConceptNet* [26] also provides assertions in triplet: a pair of words or expressions linked by a relation. In this multilingual resource, we verify if an English-French pair is directly linked; indirectly linked by another French segment or simply not linked.[7] Three forms are tested: original form, lemmatized form and lemmatized filtered form.[8]

8) On the lemmatized filtered form, we calculate the percentage of tokens which are linked with a relation of derivation, based on the resource *ConceptNet*. For example *deceptive* and *illusion* 'illusion' are not directly linked in the resource, but they are both linked to *illusoire* 'illusory'. Hence

[5]The tags of content words include: ADJ, ADV, NOUN, PROPN, VERB. If a segment does not contain any content word, the original segment is used.

[6]The lemmatization is done by *Stanford CoreNLP* and *Tree Tagger* [29] for English and French.

[7]The EN-FR and FR-FR assertions are used in this work.

[8]We filter the words in a manual list, for example the light verbs, determinants, pronouns, *etc.*

ISSN 2395-8618

we consider that there exists a link of derivation between them.

For the three following features, we have exploited the lexical translation probability table generated by the statistical word alignment tool *Berkeley Word Aligner* [30], trained on an English-French parallel corpus composed of TED Talks and a part of Paracrawl corpus (in total 1.8M parallel sentence pairs and 41M English tokens).[9]

9) The entropy of the distributions of lexical translation probabilities [31], [17], calculated according to this equation: $H(X) = \sum_i P(x_i)I(x_i) = -\sum_i P(x_i)log_e P(x_i)$. We calculate the average entropy on content words. A bigger entropy indicates that the words have more general meanings or they are polysemous. The same feature is calculated on the lemmatized content words.

10) The bidirectional lexical weighting on content words, by supposing a *n-m* alignment $a$ between the segments ($\bar{e}$ and $\bar{f}$). In the scheme proposed by Koehn *et al.* [32] (equation 1), to calculate the direct lexical weighting, each of the English words $e_i$ is generated by aligned foreign words $f_j$ with the word translation probability $w(e_i|f_j)$. And similarly for the reverse lexical weighting $lex(\bar{f}|\bar{e}, a)$. The same feature is calculated for lemmatized content words. This feature could reflect the alignment confidence between a pair of segments.

$$lex(\bar{e}|\bar{f}, a) = \prod_{i=1}^{length(\bar{e})} \frac{1}{|\{j|(i,j) \in a\}|} \sum_{\forall(i,j)\in a} w(e_i|f_j) \quad (1)$$

11) The sum of lexical translation probability differences between the human translation and the most probable translation according to the probability table. For each source word, we take the target word in human translation with the biggest probability. According to this method, we also count the unaligned words to calculate a ratio on the total number of tokens on each side. These features are calculated in the two directions of translation.

We use the toolkit *Scikit-Learn* [33] to train different statistical machine learning classifiers.[10]

### B. End-to-end Neural Network Architectures

The source and target phrases are encoded using a bidirectional encoder with Gated Recurrent Unit (GRU) (size 10). The outputs of forward and backward recurrent networks are concatenated to form the source and target phrase representations (size 20). After the encoder layer we have tried two different architectures. The first one is to build an alignment matrix for the source-target phrases, using the dot product of the two representations, inspired by these two work [34], [14]. Then a *Convolutional Neural Network* (CNN) classifier is applied to this alignment matrix, which is composed of one convolution layer followed by pooling. Since the shape of the alignment matrix varies from one source-target pair to another, an adaptive pooling is used [35]. The output of the pooling layer is fed into a fully-connected layer followed by a linear layer as the output. In the second architecture, the source and target outputs of the encoder layer are averaged over time steps to produce two fixed-dimensional vectors, which are further concatenated (size 40) and fed into a Multi-layer Perceptron (MLP) classifier. The hidden layer of MLP includes 10 hidden units with tanh non-linearity.

The length of our phrases is usually short, especially for word-for-word *Literal* instances. In order to build a more robust alignment matrix and to avoid the out-of-vocabulary problem, we finally choose to use character embeddings. As shown in table II, for the embedding layer, we have tried respectively randomly initialized character embeddings (size 10), and training our own word embeddings using skipgram model of *FastText* [36] on a TED Talks corpus (around 3M tokens for both English and French), with a word-embedding size of 100, minimum n-gram of 3, and maximum n-gram of 6. All the models have been trained in 200 epochs, with a learning rate of 0.0001 using Adam optimizer and the minibatch size of 20. Dropout has been applied to all layers except the output and embedding layers.

## V. Experimental Results and Analysis

Table II and III show the results of our classifiers using end-to-end neural network architectures, for binary classification (balanced distribution) and multi-class classification. For the binary classification, *Non_literal* (NL) class has in total 1127 instances, and 1127 *Literal* (L) instances are randomly chosen. Besides the preprocessing steps of lowercasing and correcting minor spelling errors, for the neural classifiers, we also normalized the clitic forms to complete words (*e.g. 're → are*), and normalized digits to letter form (*e.g. 42 → four two*). The architecture using word embeddings and MLP obtain better results and is faster than the other two architectures. However, the current data set is too small for neural architectures to produce satisfactory results.

TABLE II
Binary classification
(balanced distribution)

| Architecture | Accuracy | F1 (L) | F1 (NL) |
|---|---|---|---|
| Randomly initialized character embedding | | | |
| CNN | 59.99% | 0.60 | 0.60 |
| MLP | 71.16% | 0.71 | 0.71 |
| Pre-trained fasttext word embedding | | | |
| MLP | **71.25%** | 0.71 | 0.71 |

The number of all non-literal instances (1127) is only one third of *Literal* instances (3771). Considering this important difference, for the statistical machine learning classifiers, we first evaluated them under these configurations:

- six classes (*Literal, Equivalence, Generalization, Particularization, Modulation, Contain_Transposition*). We first put

[9]https://wit3.fbk.eu/, https://paracrawl.eu/index.html

[10]The code and data set is publicly available at https://github.com/YumingZHAI/ctp.

ISSN 2395-8618

TABLE III
MULTI-CLASS CLASSIFICATION
(FIVE NON-LITERAL CLASSES)

| Architecture | Accuracy | Micro-F1 | Macro-F1 |
|---|---|---|---|
| Randomly initialized character embedding | | | |
| CNN | 34.08% | 0.34 | 0.20 |
| MLP | 40.74% | 0.41 | 0.34 |
| Pre-trained fasttext word embedding | | | |
| MLP | **43.22%** | 0.43 | 0.34 |

all *Literal* instances. Then to have an approximately balanced class distribution, we randomly take 200 instances for *Literal*.

- two classes (*Literal* and *Non_literal*), with three distributions (3:1, 2:1, 1:1). The distribution 3:1 is the natural distribution in the data set. The instances of *Literal* have been extracted randomly for the last two distributions.

- five classes (only non-literal categories).

For each configuration, we have tuned the hyperparameters of different classifiers. We evaluate them by five-fold cross-validation,[11] using the metrics such as the average accuracy of five folds, the micro average and macro average F1-score [37]. The *DummyClassifier* is used as a baseline, which generates random predictions by respecting the distribution of training classes.

First, we attempted a direct classification into six classes (see table IV). The best results by *RandomForest* reflect the difficulty of the task in multi-classes. On the other hand, we observe the potential of our features on classifying the category *Literal* when the number of instances increases. As a result, we decide to divide the problem: conduct first a binary classification, and secondly a multi-class classification among the non-literal categories.

For the binary classification, the two best classifiers are *RandomForest* and MLP. Furthermore, *RandomForest* has better performance than the two combined by the method *hard voting* or *soft voting*. The table IV presents the results under three different class distributions. From the natural distribution (3:1) to our artificial balanced distribution by randomly choosing *Literal* instances (thus both class have 1127 instances), the average F1-score for the class *Non_literal* increases from 0.78 to 0.88. We will continue to test this tendency when a larger data set is available. Table IV also shows the results for the classification into five non-literal classes using all features, and the average F1-score for each non-literal category are shown in table V. The category *Generalization* has many fewer instances than the other categories, which need to be augmented; there exist many confusions between *Modulation* and the other categories, which suggests rather a review of annotation guidelines.

Table VI recapitulates the best performance on binary classification (balanced distribution) and on the classification of five non-literal classes, using the most helpful set of features. With the best performing classifier *RandomForest*, we have investigated the performance of features one by one and also grouped them: *PoS_tagging* (feature 1, 2), *surface* (feature 3), *syntactic_analysis* (feature 4, 5), *external_resource* (feature 6, 7, 8) and *word_alignment* (feature 9, 10, 11). For binary classification, feature 10 (bidirectional lexical weighting) is most helpful, which generates average F1-score of 0.78 for *Literal* and 0.80 for *Non_literal* by itself. The group of features *word_alignment* contributes the most for the binary classification. The combination of all features generates the best results, which remain the same if we remove the feature 4 (constituency parsing), 7 (how the pair is linked in the resource *ConceptNet*) and the features on PoS tagging apart from the vector counting the occurrence of each tag. The features in float form generally perform better than those in discrete form (*e.g.* 0, 1, *etc.*). Concerning the classification into five non-literal classes, the combination of all features except the group *external_resource* leads to the best results, where the group *PoS_tagging* and *syntactic_analysis* contribute more than the group *word_alignment* and *surface*. The accuracy changes from 55.10% to 55.20% after feature ablation (see table IV).

Our error analysis shows that in binary classification, it is difficult to distinguish *Literal* and *Equivalence*; in multi-class classification, the biggest confusion is between *Equivalence* and *Contain_Transposition*. Consequently, we conducted another three binary classification experiments (see table VII), where in all configurations each class has 549 instances to make the results comparable: i) *Literal* vs *Non_literal* ii) *Literal* combined with *Equivalence* (E), vs the other classes iii) *Literal* combined with *Equivalence* and *Transposition* (T), vs the other classes. The third configuration is more interesting, because the group of translation processes *LET* do not bring meaning changes, while the processes *non-LET* could. The results show that by including *Transposition* (change grammatical classes without changing the meaning), the performance gets better than only grouping *Literal* and *Equivalence*, since we avoid the confusion between *Equivalence* and *Transposition*. The better results of binary classification (L vs NL, LET vs non-LET) indicate that in future work we can develop cascading classifiers, namely first separating word-for-word literal translations, or those which do not cause meaning changes, then conducting a finer-grained classification among the other categories.

## VI. CONCLUSION AND PERSPECTIVES

We have proposed a new Natural Language Processing task of automatically classifying translation processes at subsentential level, based on manually annotated examples from a parallel English-French TED Talks corpus. To the best of our knowledge, these translation processes have not been explicitly exploited during paraphrase extraction from bilingual parallel corpora. With the best performing classifier *RandomForest* and feature engineering, our empirical results show a best accuracy of 87.09% for

[11] StratifiedKFold is used for cross-validation, where the folds are made by preserving the percentage of samples for each class.

ISSN 2395-8618

TABLE IV
CLASSIFICATION RESULTS UNDER DIFFERENT CONFIGURATIONS, USING ALL FEATURES

| Distribution of classes | Classifier | Accuracy | Micro-F1 | Macro-F1 |
|---|---|---|---|---|
| **Six classes** | | | | |
| six classes, with 3771 *Literal* | Dummy | 60.76% | 0.61 | 0.15 |
| | RandomForest | **83.10%** | 0.83 | 0.44 |
| six classes, with 200 *Literal* | Dummy | 18.92% | 0.19 | 0.16 |
| | RandomForest | 57.04% | 0.57 | 0.52 |
| **Two classes** | | | | |
| *Literal* (3) : *Non_literal* (1) | Dummy | 65.84% | 0.66 | 0.52 |
| | RandomForest | **90.16%** | 0.90 | 0.86 |
| *Literal* (2) : *Non_literal* (1) | Dummy | 56.43% | 0.56 | 0.51 |
| | RandomForest | 88.85% | 0.89 | 0.88 |
| *Literal* (1) : *Non_literal* (1) | Dummy | 53.19% | 0.53 | 0.53 |
| | RandomForest | **87.09%** | 0.87 | 0.87 |
| **Five classes** | | | | |
| Five non-literal classes | Dummy | 20.32% | 0.20 | 0.18 |
| | RandomForest | **55.10%** | 0.55 | 0.47 |

TABLE V
AVERAGE F1-SCORE FOR EACH NON-LITERAL CLASS, USING ALL FEATURES

| Category | Equivalence | Generalization | Particularization | Modulation | Contain_Transposition |
|---|---|---|---|---|---|
| Nb. instances | 289 | 86 | 215 | 195 | 342 |
| Average F1 | 0.51 | 0.25 | 0.56 | 0.36 | 0.68 |

TABLE VI
CLASSIFICATION RESULTS AFTER FEATURE ABLATION STUDY

| | average accuracy | average F1-scores | |
|---|---|---|---|
| binary classification (balanced distribution) | **87.09%** | 0.87 (Literal) | 0.88 (Non_literal) |
| five non-literal classes | **55.20%** | 0.55 (micro average) | 0.48 (macro average) |

TABLE VII
CLASSIFICATION RESULTS AFTER GROUPING CLASSES, EVERY CLASS HAS 549 INSTANCES

| Configuration | average accuracy | average F1 (class1) | average F1 (class2) |
|---|---|---|---|
| Dummy | 48.63% | 0.49 | 0.49 |
| L vs NL | 85.24% | 0.84 | 0.86 |
| LE vs non-LE | 75.32% | 0.74 | 0.77 |
| LET vs non-LET | 79.42% | 0.78 | 0.81 |

binary classification (*Literal* vs *Non_literal*) and 55.20% for multi-class classification (*Equivalence, Generalization, Particularization, Modulation, Contain_Transposition*), which are much better than the baseline random classifier.

This task is complicated, and our exploratory work is restrained by the limited amount of annotated examples. However, our work demonstrates that automatically classifying translation processes seem possible, and the experiments show the directions that we can follow in future work. There is much room to constitute an augmented and balanced data set, on which we will evaluate our classifier to observe the performance. The finer error analysis of the classification results is useful to help the research on corpus annotation and linguistic analysis. We will continue to improve the classifier on English-French, by implementing other features for multi-class classification, and explore more neural architectures. We will also extend our work to English-Chinese translation pairs. One of our long term objectives is leveraging this automatic classification to better control paraphrase extraction from bilingual parallel corpora.

## REFERENCES


[1] J.-P. Vinay and J. Darbelnet, *Stylistique comparée du français et de l'anglais: méthode de traduction*, ser. Bibliothèque de stylistique comparée. Didier, 1958.

[2] P. Newmark, *Approaches to Translation (Language Teaching Methodology Senes)*. Oxford: Pergamon Press, 1981.

[3] H. Chuquet and M. Paillard, *Approche linguistique des problèmes de traduction anglais-français*. Ophrys, 1989.

[4] L. Molina and A. Hurtado Albir, "Translation techniques revisited: A dynamic and functionalist approach," *Meta*, vol. 47, no. 4, pp. 498–512, 2002.

[5] B. J. Dorr, L. Pearl, R. Hwa, and N. Habash, "Duster: A method for unraveling cross-language divergences for statistical word-level alignment," in *Conference of the Association for Machine Translation in the Americas*. Springer, 2002, pp. 31–43.

[6] D. Deng and N. Xue, "Translation divergences in Chinese–English machine translation: An empirical investigation," *Computational Linguistics*, vol. 43, no. 3, pp. 521–565, 2017.

[7] C. Bannard and C. Callison-Burch, "Paraphrasing with bilingual parallel corpora," in *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 2005, pp. 597–604.

ISSN 2395-8618


[8] J. Mallinson, R. Sennrich, and M. Lapata, "Paraphrasing revisited with neural machine translation," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, vol. 1, 2017, pp. 881–893.

[9] J. Ganitkevitch, B. Van Durme, and C. Callison-Burch, "PPDB: The paraphrase database," in *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2013, pp. 758–764.

[10] E. Pavlick, J. Bos, M. Nissim, C. Beller, B. Van Durme, and C. Callison-Burch, "Adding semantics to data-driven paraphrasing," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, vol. 1, 2015, pp. 1512–1522.

[11] Y. Zhai, A. Max, and A. Vilnat, "Construction of a multilingual corpus annotated with translation relations," in *First Workshop on Linguistic Resources for Natural Language Processing*, 2018, pp. 102–111.

[12] M. Carpuat, Y. Vyas, and X. Niu, "Detecting cross-lingual semantic divergence for neural machine translation," in *Proceedings of the First Workshop on Neural Machine Translation*. Association for Computational Linguistics, 2017, pp. 69–79.

[13] Y. Vyas, X. Niu, and M. Carpuat, "Identifying semantic divergences in parallel text without annotations," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, M. A. Walker, H. Ji, and A. Stent, Eds. Association for Computational Linguistics, 2018, pp. 1503–1515.

[14] M. Q. Pham, J. Crego, J. Senellart, and F. Yvon, "Fixing translation divergences in parallel corpora for neural mt," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 2967–2973.

[15] P. Newmark, *A textbook of translation*. Prentice Hall New York, 1988, vol. 66.

[16] K. Imamura, E. Sumita, and Y. Matsumoto, "Automatic construction of machine translation knowledge using translation literalness," in *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics-Volume 1*. Association for Computational Linguistics, 2003, pp. 155–162.

[17] M. Carl and M. J. Schaeffer, "Why translation is difficult: A corpus-based study of non-literality in post-editing and from-scratch translation," *HERMES-Journal of Language and Communication in Business*, no. 56, pp. 43–57, 2017.

[18] R. Chatterjee, M. Negri, R. Rubino, and M. Turchi, "Findings of the WMT 2018 shared task on automatic post-editing," in *Proceedings of the Third Conference on Machine Translation*. Belgium, Brussels: Association for Computational Linguistics, October 2018.

[19] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul, "A study of translation edit rate with targeted human annotation," in *Proceedings of association for machine translation in the Americas*, vol. 200, no. 6, 2006.

[20] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pp. 311–318.

[21] J. Cohen, "A coefficient of agreement for nominal scales," *Educational and Psychological Measurement*, vol. 20, pp. 37–46, 1960.

[22] S. Petrov, D. Das, and R. T. McDonald, "A universal part-of-speech tagset," in *Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012, Istanbul, Turkey, May*

[25] M. Candito, J. Nivre, P. Denis, and E. H. Anguiano, "Benchmarking of statistical dependency parsers for french," in *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, *23-25, 2012*, N. Calzolari, K. Choukri, T. Declerck, M. U. Dogan, B. Maegaard, J. Mariani, J. Odijk, and S. Piperidis, Eds. European Language Resources Association (ELRA), 2012, pp. 2089–2096.

[23] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky, "The Stanford CoreNLP natural language processing toolkit," in *Association for Computational Linguistics (ACL) System Demonstrations*, 2014, pp. 55–60.

[24] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Soviet physics doklady*, vol. 10, no. 8, pp. 707–710, 1966.

Association for Computational Linguistics. Chinese Information Processing Society of China, 2010, pp. 108–116.

[26] R. Speer, J. Chin, and C. Havasi, "Conceptnet 5.5: An open multilingual graph of general knowledge," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017, pp. 4444–4451.

[27] J. Camacho-Collados, M. T. Pilehvar, N. Collier, and R. Navigli, "Semeval-2017 task 2: Multilingual and cross-lingual semantic word similarity," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Association for Computational Linguistics, 2017, pp. 15–26.

[28] R. Speer and J. Lowry-Duda, "ConceptNet at SemEval-2017 task 2: Extending word embeddings with multilingual relational knowledge," in *Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval@ACL 2017, Vancouver, Canada, August 3-4, 2017*, S. Bethard, M. Carpuat, M. Apidianaki, S. M. Mohammad, D. M. Cer, and D. Jurgens, Eds. Association for Computational Linguistics, 2017, pp. 85–89.

[29] H. Schmid, "Improvements in part-of-speech tagging with an application to German," in *Proceedings of the ACL SIGDAT-Workshop*, 1995, pp. 47–50.

[30] P. Liang, B. Taskar, and D. Klein, "Alignment by agreement," in *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*. Association for Computational Linguistics, 2006, pp. 104–111.

[31] R. M. Gray, *Entropy and Information Theory*. Berlin, Heidelberg: Springer-Verlag, 1990.

[32] P. Koehn, F. J. Och, and D. Marcu, "Statistical phrase-based translation," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*. Association for Computational Linguistics, 2003, pp. 48–54.

[33] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[34] J. Legrand, M. Auli, and R. Collobert, "Neural network-based word alignment through score aggregation," in *Proceedings of the First Conference on Machine Translation, WMT 2016, colocated with ACL 2016, August 11-12, Berlin, Germany*. The Association for Computer Linguistics, 2016, pp. 66–73.

[35] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, Sept 2015.

[36] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[37] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Random k-labelsets for multilabel classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 7, pp. 1079–1089, 2011.