Comparing a Hand-crafted to an Automatically Generated Feature Set for Deep Learning: Pairwise Translation Evaluation

Despoina Mouratidis
Department of Informatics, Ionian University
/ Tsirigoti Squ. 7, 49100 Corfu, Greece
c12mour@ionio.gr

Katia Lida Kermanidis
Department of Informatics, Ionian University
/ Tsirigoti Squ. 7, 49100 Corfu, Greece
kerman@ionio.gr

Abstract

The automatic evaluation of machine translation (MT) has proven to be a very significant research topic. Most automatic evaluation methods focus on the evaluation of the output of MT as they compute similarity scores that represent translation quality. This work targets on the performance of MT evaluation. We present a general scheme for learning to classify parallel translations, using linguistic information, of two MT model outputs and one human (reference) translation. We present three experiments to this scheme using neural networks (NN). One using string based hand-crafted features (Exp1), the second using automatically trained embeddings from the reference and the two MT outputs (one from a statistical machine translation (SMT) model and the other from a neural machine translation (NMT) model), which are learned using NN (Exp2), and the third experiment (Exp3) that combines information from the other two experiments. The languages involved are English (EN), Greek (GR) and Italian (IT) segments are educational in domain. The proposed language-independent learning scheme which combines information from the two experiments (experiment 3) achieves higher classification accuracy compared with models using BLEU score information as well as other classification approaches, such as Random Forest (RF) and Support Vector Machine (SVM).

1 Introduction

MT systems need to be evaluated in order to assess the degree of reliability of their results, and to facilitate means for improvement as well. Some of the most popular automatic MT evaluation methods are the BLEU score (Papineni et al., 2002), TER (Snover et al., 2006), METEOR (Lavie and Agarwal, 2007) etc. Zechner and Waibel 2000 introduced the word error rate (WER), a lexical similarity metric. WER uses the number of steps required to make the output similar to reference translation. Mouratidis and Kermanidis (2019) used parallel corpora and they showed that string-based features (e.g. length of source (*src*) sentence), similarity based (e.g. the ratio of common suffix of MT outputs and the reference) etc. could improve the performance of MT system. Giménez and Màrquez (2007) used syntactic similarity methods like information from part of speech tagging (POS). Pighin and May (2012) proposed the analysis of an annotated corpus based on automatic translation and user-provided translation corrections gathered through an online MT system. Barrón-Cedeño et al. (2013) used an extension of the corpus of the study by Pighin and May (2012). They introduced new features and they tried different configurations of classifiers. Both papers showed that the quality of an SMT system can be improved.

Word representations (embeddings) are very useful in Natural Language Processing (NLP) applications such as automatic speech recognition and MT (Schwenk, 2007). They can model the semantic and syntactic information of every word in a document (Hill et al., 2014). There are lots of different methods for generating embeddings such as methods based on simple recursive neural networks (RNN) (Cho et al., 2014), convolutional neural networks and RNN using Long short-term memory (LSTM) (Sutskever et al., 2014), count-based methods and others. A big variety of pre-trained embedding models are used in the literature, such as Word2Vec (Mikolov et al., 2014) and GloVe (Pennington et al., 2014).

Temnikova, I, Orăsan, C., Corpas Pastor, G., and Mitkov, R. (eds.) (2019) *Proceedings of the 2nd Workshop on Human-Informed Translation and Interpreting Technology (HiT-IT 2019)*, Varna, Bulgaria, September 5 - 6, pages 66–74.
https://doi.org/10.26615/issn.2683-0078.2019_008

Because of the wide spread development of DL techniques, many researchers have utilized neural networks for MT evaluation. Duh (2008) uses a learning framework for ranking translations in parallel settings, given representations of translation outputs and a reference translation. Duh (2008) used a feature set containing some simple string-based features, like length of the words, but also BLEU score information. He used ranking-specific features and he showed that ranking achieves higher correlation to human judgments. Another important work is presented by Guzmán et al. (2015), (2017) who integrated syntactic and semantic information about the reference and the machine-generated translation as well, by using pre-trained embeddings and the BLEU scores of the translations. They used a multi-layer NN to decide which of the MT outputs is better. Ma et al. (2016) designed metrics based on LSTM, allowing the evaluation of single hypothesis with reference, instead of pairwise situation.

In this paper, we consider the choice of the best translation as a classification problem to be solved using deep learning architectures, by investigating two translation prototypes for our experiments. One is based on SMT and the other on NMT. We present a general learning scheme to classify machine-generated translations, using information from linguistic representations and one reference translation, for two language pairs (EN-GR, EN-IT). Unlike earlier works, the present approach includes the following novelties:

- Automatically extracted embeddings in two languages: GR and IT.
- A learning scheme based on a combination of a hand-crafted feature set (string similarity) and automatically trained embeddings as well.
- The proposed approach is language-independent.

To the author's knowledge, this is the first time that this architecture is used for a classification task using automatically extracted embeddings and hand-crafted features for this particular data genre, and these language pair.

The rest of the paper is organized as follows: Section 2 describes the corpora, the feature set (hand-crafted features), the embeddings, the annotation procedure and the experimental setup. Section 3 presents and analyzes our experimental results (including linguistic analysis). Finally, section 4 presents our conclusions and directions for future research.

2 Materials and Methods

2.1 Data

The dataset used in our work is a parallel corpus which is part of the test sets developed in the TraMOOC project (Kordoni et al., 2016). The corpora consist of educational data, lecture subtitle transcriptions etc., with unorthodox syntax, ungrammaticalities etc (i.e. 1.To criticize, 2. Has no objections.). The corpora are described in detail by Mouratidis and Kermanidis (2018), (2019). The EN-GR corpus consists of 2686 sentences, whereas the EN-IT corpus of 2745 sentences. For each sentence, two translations were provided, generated by the Moses phrase-based SMT toolkit (*T1*) (Koehn et al., 2007) and the NMT Nematus toolkit (*T2*) (Sennrich, 2017). Moreover, a professional translation (*Tr*) is provided and used as a reference for each language. Both models are trained on both in- and out- of domain data. Out-of-domain data included corpora e.g., Europarl, WMT News corpora etc. In-domain data included data from TED, Coursera, etc. (Barone et al., 2017). NMT model is trained on additional in-domain data provided via crowdsourcing. More details on the datasets can be found in Sosoni et al. (2018).

2.2 Annotation

We consider the translation evaluation problem as a binary classification task. Two MT outputs *T1* and *T2* and the reference segment (*Tr*) are provided. Two annotators, for each language pair, annotated the corpora, as follows:

$$y = \begin{cases} 0, if\ T1\ is\ worse\ than\ T2 \\ 1, if\ T1\ is\ better\ than\ T2 \end{cases} \quad (1)$$

In order to decide if *T1* is better than *T2*, annotators used the source and reference sentences. The two annotators had an inter-annotator disagreement percentage of 3% for EN-GR and 5% for EN-IT. For the different answers, the annotators discussed and agreed on one class. The ID3 on Table 3 (Appendix) is an example of disagreement for the EN-GR language pair. We observed low annotation value for SMT class (38% EN-GR

/ 43% EN-IT) compared with NMT class (62% EN-GR /57% EN IT).

2.3 Features

We decided to use linguistic features based on string similarity, that involve no morphosyntactic information (no information about word forms and sentence structure), and are language independent. The features used were (i) Simple features (e.g. length in tokens, or some distances), (ii) Noise-Based features (e.g. frequentness of the repeated words) and (iii) Similarity-Based features (e.g. character 3-gram similarity). Each segment pair (*T1,Tr*), (*T2,Tr*) was modeled as a feature-value vector. The features have values between 0 and 1. The feature set was based on the work described in Mouratidis and Kermanidis (2019), with the difference that we used two classes instead of three (one for every MT output). This reduction in the number of classes was performed in order to allow for a more straightforward comparison between the three experiments and related work.

2.4 Word Embeddings

Word embeddings are very important in our model, because they allow us to model the relations between the two translations and the reference. In this work, we created and trained our own embeddings between the two MT outputs, as well as the reference translation for the two target languages (GR and IT). To prepare the input to the embedding layer, we used the bag of words model encoding a one hot function to generate the integer matrix. In order to avoid the inputs having different lengths, we used the pad sequences function, which padded all inputs to have the same length. The size, in number of nodes, of the embedding layer is 64 for both languages. The input dimensions of the embedding layers are in agreement with the vocabulary of each language (taking into account the most frequent words): 400 for EN-GR and 200 for EN-IT. We used the embedding layer provided by Keras (Chollet, 2015) with TensorFlow as backend (Abadi et al., 2016).

2.5 Experimental Setup

Experiment 1: For the first experiment, we used tuples (*T1*, *T2*, *Tr*), with string based features (the 2D matrix *A[i,j]*). Matrix *A[i,j]* contains 50 hand-crafted linguistic features (described in section 2.3) for every segment based on Mouratidis and Kermanidis (2019), where *i* represents the number of segments (*T1*,*T2*,*Tr*) and *j* the number of features. In this work, we have used two classification classes (one for the SMT output and the other for the NMT output) instead of three (used in Mouratidis and Kermanidis (2019)). Furthermore, a different network architecture is used, a simple but classic architecture of three Dense (Feed-Forward) layers. Dense layers serve the purpose of doing the classification. We also used a dropout layer to every Dense layer to prevent overfitting. Also a NN API is used instead of the WEKA framework (used in Mouratidis and Kermanidis (2019)). The model architecture used for the first experiment is shown in Fig.1.

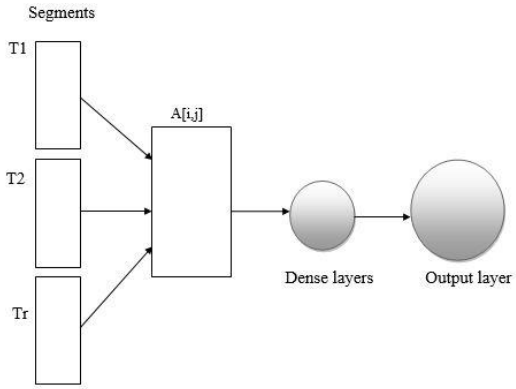

Figure 1: Learning scheme for Exp1.

Experiment 2: Based on the sentences *T1*, *T2*, *Tr*, we have created word embeddings (*EmbT1*, *EmbT2*, *EmbTr*). We used the word embeddings to find the probability for segment *T1* to be better than *T2* and vice-versa, given *Tr* and *y*. The probability is a Bernoulli conditional distribution (Krstovski and Blei, 2018).

$$p(y/T1,T2,Tr) = Bernoulli(y/b) \quad (2)$$

The parameter b_y is defined as follows:

$$b_y = \sigma\left(w^T f(T1,T2,Tr)\right) \quad (3)$$

where σ is the sigmoid function, w^T are the rows of a weight matrix *W*, and function *f* is the transformation of *T1*, *T2* and *Tr* in the hidden layer, i.e. *f(T1,T2,Tr)=[h1,h2,hr]*. The embeddings for every tuple (*T1*, *T2*, *Tr*) are concatenated in a pairwise fashion, i.e. i. *EmbT1,EmbT2*, ii. *EmbT1,EmbTr*, iii. *EmbT2,EmbTr*. These fixed-length vectors are the input for the evaluation groups *h12*, *h1r*, *h2r*. We have checked if *T1* and *T2* are similar to the reference translation *Tr*

(*h1r, h2r* respectively), but also if *T1* is similar to *T2* (*h12*). This is quite interesting because in many cases we observed a similarity between *T1* and *T2*, but this does not mean that they were the proper translations, when compared to *Tr*. The input to our neural model is represented by concatenating the vector representation of the outputs of these evaluation groups.
The model architecture used for the second and the third experiment is shown in Fig.2.

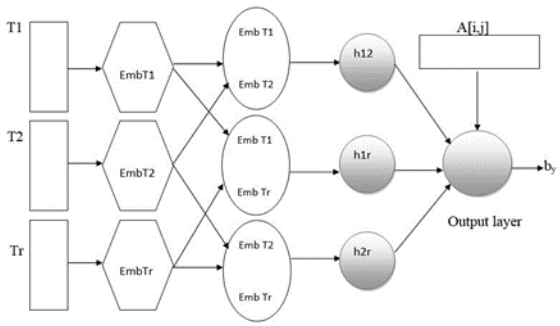

Figure 2: Learning scheme for Exp2 and 3**.**

Experiment 3: In this experiment, we utilized the tuple (*T1, T2, Tr*) as input to our model and the same configuration with Exp2 as well. We wanted to find out if the hand-crafted features, in combination with the automatically extracted embeddings, can improve classification accuracy of Exp2. For this purpose, as an extra input to our neural model, we utilized the 2D matrix *A[i,j]* with hand-crafted features (string-based), described in the Exp1.

Particularly, the model architecture for the first experiment is defined as follows:

- Size of layers: Dense 1 & 2 with 128 Hidden Units, Dense 3 with 64 Hidden Units
- Output layer: Activation Sigmoid
- Learning rate: 0.001
- Activation Function of Dense Layers: Softmax
- Dropout of Dense Layers: 0.2
- Lossfunction: Binary cross entropy

The architecture for the second and third experiments is a classic architecture of Dense (Feed-Forward) layers. After running multiple tests, we configured our experiments as follows:

- Size of layers: Dense 1, 2 &3 with 128 Hidden Units, (Dense 4 with 64 Hidden Units)
- Activation Function of Dense Layers: Relu
- Dropout of Dense Layers: 0.4

The networks are trained using the stochastic optimizer Adam (Kingma and Lei Ba, 2014) with a learning rate of 0.005. In Table 1, we present the complete set training parameters.

	Exp1	**Exp2**	**Exp3**
Batch size	128	64	256
Epochs	5	30	10

Table 1: Training parameters for Exp2/Exp3.

As a validation option for all the experiments, we used 10 fold cross validation (CV), which is effective for small datasets.

3 Results

In this section, we present the results from our experiments. We utilized the Positive Predictive Value (Precision) and the Sensitivity (Recall), as evaluation metrics, which are commonly used in classification tasks. The first metric shows which proportion of identifications is actually correct, whereas the second metric shows that the proportion of actual positives is correctly identified.

Fig. 3 and Fig. 4 show the accuracy performance of our experiments for both classes (SMT, NMT) for EN-GR and EN-IT respectively.

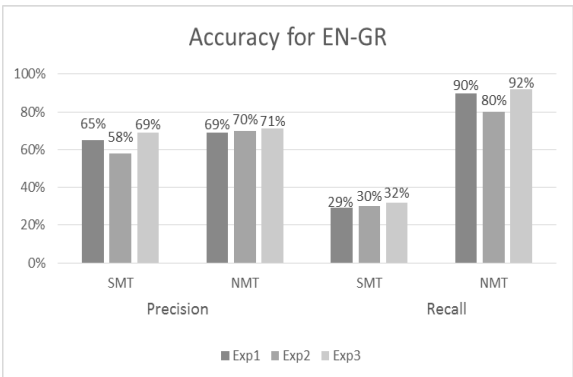

Figure 3: Accuracy for EN-GR.

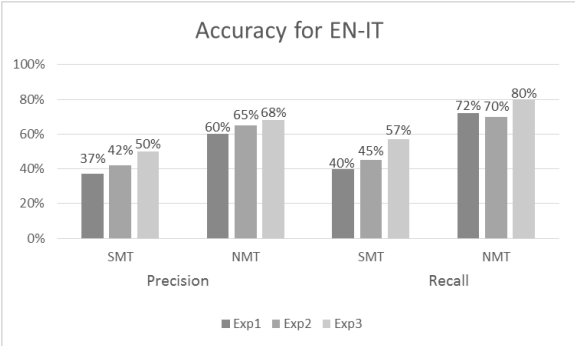

Figure 4: Accuracy for EN-IT.

We observed that the use of hand-crafted features in combination with embeddings have a positive effect on performance for both language pairs. Table 2 shows the accuracy results between our proposed model and the model proposed by Guzmán et al. (2017) which uses information from the MT evaluation metric BLEU score for language pairs EN-GR and EN-IT. The BLEU metric does not distinguish between content and function words and it is a language independent metric. As we are dealing with an uneven class distribution, unbalanced scores between Precision and Recall are observed, we present the F1 Score as well. F1, in statistical analysis of binary classification, is a measure of a test's accuracy. It penalizes classifiers with imbalanced precision and recall scores (Chinchor, 1992). As an averaging method, we used macro average.

Our proposed model (Exp3) achieved better accuracy performance than the model using information from Bleu scores of the MT outputs. A reason for that may be that BLEU attempts to measure the correspondence between an MT output and a human translation. Nevertheless, the hand-crafted feature set provides more information about not only the correspondence but also the correlation between suffixes, word distances and others.

To enable a direct comparison of our experimental results with earlier work (Barrón-Cedeño et al., 2013, Mouratidis and Kermanidis, 2019), we ran additional experiments using the WEKA framework as backend (Singhal and Jena, 2013). Different configurations were experimented with, including SVM and RF for EN-GR (Fig. 5) and EN-IT (Fig. 6).

	AVG Precision	**AVG Recall**	**AVG F1**
Language pair EN-GR			
Hand-crafted features + embeddings (Exp3)	69%	69%	65%
Bleu score	63%	63%	60%
Language pair EN-IT			
Hand-crafted features + embeddings (Exp3)	62%	68%	64%
Bleu score	60%	60%	62%

Table 2: Comparison with Bleu score.

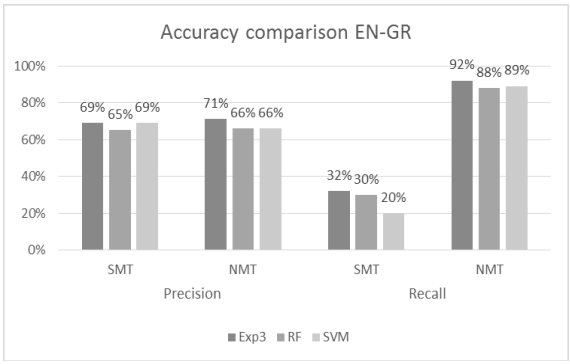

Figure 5: Accuracy comparison with other approaches for EN-GR**.**

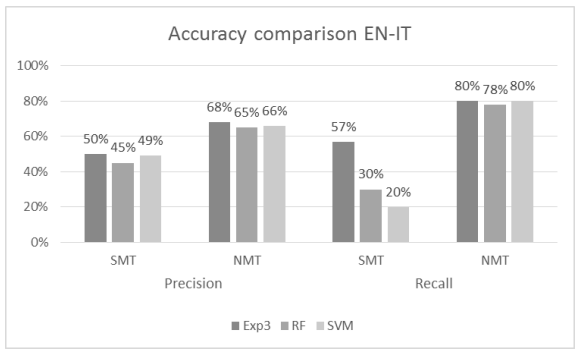

Figure 6: Accuracy comparison with other approaches for EN-IT.

We conclude that all evaluation metrics demonstrated the primacy of the NMT model over the SMT one, which agrees with the annotators' choice.

3.1 Linguistic Analysis

In order to show a part of the linguistic reasons for these accuracy values, we show some segments (*T1, T2, Tr*) from the EN-GR parallel corpus (Table 3, Appendix). ID 3 is an example of annotation disagreement.

For ID1:

•*T2* has erroneously translated the word *hoods* as *κουκκίδες*: *dots*, instead of the correct and most common translation: *κουκούλες*. There is no obvious and understandable reason for this.

•The word *flagella* is a really problem in *T1*, *T2* and *Tr* sentences. The word is of Latin origin (*flagellum*, diminutive of *flagrum*: *whip*) and has not been changed in English. In science, there is the bacterial flagellum, translated in Greek as: *μαστίγιο των βακτηρίων*. *T1* did not at all translate it, *T2* translated it as *πλάκες μαστιγίων* (*plates of whips*), but the *Tr* as *βλεφαρίδες* (*eyelashes*).

•*T1* has chosen the most common translation for the word *nodules*: *οζίδια*, but according to the *Tr*, the word has probably the sense of *clots*. On the contrary, *T2* has translated the word as *ακίδες*: *pins, thorns, splinters*.

•*T1* and *T2* have correctly identified the sense of the word *stems* as *κοτσάνια* and *μίσχων* respectively. Nevertheless, these two words are not used in the same contexts. *Κοτσάνι* is a hellenized slavic word, commonly used in oral speech, but *μίσχος* is an hellenistic word, rather used in official texts.

•Neither *T1* nor *T2* correctly translated the idiom: *and what have you*, meaning: *and many other such things, and so on, etc*. They both literally translated this expression as: *και τι έχετε* and *και σε αυτό που έχετε* respectively, meaning: *what you have got*.

In this case, the annotators have chosen T2 as the best translation, whereas Exp3's choice was T1.

For ID2:

•*T1* has correctly translated the title of Michel Foucault' book (*Επιτήρηση και Τιμωρία*). It's obvious that this title is included in *T1*'s ¨armory¨. On the contrary, *T2* translated the title in a completely wrong way, especially the second word: *Στην πειθαρχι ? α και στο Πω ? νητο*. The choice of the question marks it is not understandable.

•In *T1*, the author's name and surname haven't been translated into Greek and that is the best choice. In *T2*, these are hellenized, but the surname in a very wrong way: *Φουκούλτ*, instead of: *Φουκώ*. The second syllable of this surname has been wrongly hellenized letter by letter, without being taken into account that, according to its pronunciation, the French suffix *–ault* has been commonly hellenized: *-ω*.

•*T1* has correctly translated the word power as *εξουσίας* and not: *δύναμης*, as *T2* did. *T1* ¨knew¨ what is commonly known, that is the word *power* in the phrase: *instrument of power* has the sense of authority.

Both annotators and Exp3 have chosen T1 as the best translation.

For ID3:

Annotator 1 labeled *T1* as the better translation for the following reasons:

•Only *T1* successfully translated the ¨difficult¨ word of the text: *sumo*, as it is usually said in Greek: *σούμο*. The difficulty about this word is due to two reasons: i. The word *sumo* isn't an English word, but a Japanese one (meaning: *to compete*). ii. The same word is a paronym of the English common, well known, word: *sum*, (having, of course, a different meaning: *amount, total, aggregate*). On the contrary, *T2* ¨fell into the trap¨ of the paronym and translated the word as *a sum* (*αθροίσματος*).

•Only *T1* successfully translated the other ¨difficult¨ word of the text: *delicious* as *υπέροχα.* The problem is about the literal (*γευστικός*: *tasty*) and the figurative (*υπέροχος*: *wonderful*) sense of the word. In this segment, the word *delicious* has a figurative sense (*wonderful guys*). On the contrary, *T2* wrongly used the literal meaning (*tasty guys*!).

Annotator 2 labeled *T2* as the better translation for the following reasons:

•*T2* was the only one that has successfully translated the personal pronoun *you* (*I would like you to...*).

•*T2* has correctly translated the verb *to get* as *to obtain, to take, to collect*. The verb t*o get* is used in a lot of patterns having different meanings. One of them is: *to get*+ direct object= *to obtain*. It's just the case here: *to get fat=to fatten*. On the contrary, *T1* has wrongly translated, in a literal way, the two words (*παίρνει το λίπος*: *take the suet*!).

After discussion, the annotators finally consented to T1 as the best translation in this case, whereas Exp3 had chosen T2.

4 Conclusion and Future Work

In this paper, we have compared the handcrafted feature set with the automatically extracted

ones, for a pairwise translation evaluation application in a deep learning setting.

In particular, we ran three experiments using hand-crafted string-based features, automatically extracted embeddings and both hand-crafted string-based features and automatically extracted embeddings respectively. The purpose of our work has been to find out whether information of string-based features, in combination with embeddings, affects classification accuracy, in order to train a model which will correctly choose the best translation.

The results showed that the proposed learning scheme improved the classification accuracy when using the vector representation (word embeddings) and the hand-crafted features as well (Exp3). Additionally, we have run experiments using Bleu as extra information, as well as well-known approaches, such as RF and SVM. Our model achieved better accuracy results in all the cases. For a more integrated analysis of the accuracy results, we have also carried out a qualitative linguistic analysis.

In future work, we intend to implement other combinations of NN, layer architectures and sizes, as well as other criteria. We believe that information from the *src* sentence could improve the accuracy scores. We could experiment with other ways for calculating embeddings, for example the utilization of more sophisticated bag of word model encoding, like TF-IDF. Although there are not enough available pre-trained embeddings in languages involved in our experiments, we want to examine if the use of pre-trained embeddings will give better accuracy results. Finally, we state our willingness to improve the text preprocessing phase, as we believe that it will lead to better results.

Acknowledgments

The authors would like to thank the reviewers for their significant suggestions and comments. We would also to thank the TraMOOC project for the corpora used in our experiments.

References

Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2016. Tensorflow: A system for large-scale machine learning. In *Proceedings of 12th USENIX Symposium on Operating Systems Design and Implementation*. USENIX Association, pages 265-283.

Antonio Valerio Miceli Barone, Barry Haddow, Ulrich Germann, and Rico Sennrich. 2017. Regularization techniques for ne-tuning in neural machine translation. *In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (Volume 1: Short Papers)*. Association for Computational Linguistics, pages 1-6. http://arxiv.org/abs/1707.09920.

Alberto Barrón-Cedeño, Lluís Vilodre Màrquez, Carlos Alberto Henríquez Quintana, Lluís Fanals Formiga, Enrique Marino Romero, and Jonathan May. 2013. Identifying useful human correction feedback from an on-line machine translation service. In *Proceedings of 23rd International Joint Conference on Artificial Intelligence*. International Joint Conference on Artificial Intelligence, pages 2057–2063.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3:1137-1155.

Nancy Chinchor. 1992. MUC-4 Evaluation Metrics. *In Proccedings of the Fourth Message Understanding Conference*, pages 22–29.

Kyunghyun Cho, Bart Van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (Volume 1)*. Association for Computational Linguistics, pages 1724-1734.

François Chollet. Keras: Deep learning library for theano and tensorflow. 2015. URL: https://keras.io/k 7.8

Kevin Duh. 2008. Ranking vs. regression in machine translation evaluation. In *Proceedings of the Third Workshop on Statistical Machine Translation*. Association for Computational Linguistics, pages 191–194.

Francisco Guzmán, Joty Shafiq, Lluís Màrquez, and Nakov Preslav. 2015. Pairwise neural machine translation evaluation. 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)*. Association for Computational Linguistics, pages 805-814. doi>10.1162/COLI_a_00298.

Francisco Guzmán, Joty Shafiq, Lluís Màrquez, and Nakov Preslav. 2017. Machine translation evaluation with neural networks. *Computer Speech & Language,* 45: 180-200.

Felix Hill, Kyunghyun Cho, Sebastian Jean, Coline Devin, and Yoshua Bengio. 2015. Embedding word similarity with neural machine translation. *arXiv:1412.6448*, 4: 1-12.

Diederik Kingma, and Jimmy Lei Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 9: 1-15.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Chris Dyer, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*. Association for Computational Linguistics, pages 177-180.

Valia Kordoni, Lexi Birch, Ioanna Buliga, Kostadin Cholakov, Markus Egg, Federico Gaspari, Yota Georgakopoulou, Maria Gialama, Iris Hendrickx, Mitja Jermol, Katia Kermanidis, Joss Moorkens, Davor Orlic, Michael Papadopoulos, Maja Popović, Rico Sennrich, Vilelmini Sosoni, Dimitrios Tsoumakos, Antal van den Bosch, Menno van Zaanen, and Andy Way. 2016. TraMOOC (Translation for Massive Open Online Courses): Providing Reliable MT for MOOCs. In *Proceedings of the 19th annual conference of the European Association for Machine Translation*. European Association for Machine Translation, page 396.

Kriste Krstovski, and David M. Blei. 2018. Equation embeddings. *arXiv preprint arXiv:1803.09123*, 1: 1-12.

Alon Lavie, and Abhaya Agarwal. 2007. METEOR: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments. In *Proceedings of the Second ACL Workshop on Statistical Machine Translation*. Association for Computational Linguistics, pages 228-231.

Qingsong Ma, Fandong Meng, Daqi Zheng, Mingxuan, Yvette Graham, Wenbin Jiang and Qun Liu. 2016. Maxsd: A neural machine translation evaluation metric optimized by maximizing similarity distance. In *Natural Language Understanding and Intelligent Applications*. Springer, Cham, pages 153-161.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, & J effrey Dean. 2013. Distributed representations of words and phrases and their compositionality. *In Advances in neural information processing systems* (pp. 3111-3119).

Despoina Mouratidis, and Katia Lida Kermanidis. 2018. Automatic Selection of Parallel Data for Machine Translation. In *IFIP International Conference on Artificial Intelligence Applications and Innovations*. Springer, pages 146-156.

Despoina Mouratidis, and Katia Lida Kermanidis. 2019. Ensemble and Deep Learning for Language-Independent Automatic Selection of Parallel Data. *Algorithms*, 12(1):26. https://doi.org/10.3390/a12010026.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. 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, pages 311–318. doi>10.3115/1073083.1073135.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing*. Empirical Methods in Natural Language Processing, pages 1532-1543.

Daniele Pighin, Lluís Màrquez, and Jonathan May. 2012. An analysis (and an annotated corpus) of user responses to machine translation output. In *Proceedings of the 8th International Conference on Language Resources and Evaluation*. European Language Resources Association, pages 1131-1136.

Holger Schwenk. 2007. Continuous space language models. *Computer Speech and Language*, 21(3): 492-518. doi:10.1016/j.csl.2006.09.003.

Rico Sennrich, Orhan Firat, Kyunghyun Cho, Alexandra Birch-Mayne, Barry Haddow, Julian Hitschler, Marcin Junczys-Dowmunt, Samuel Läubli, Valerio Miceli Barone, Jozef Mokry, and Maria Nădejde. 2017. Nematus: A toolkit for neural machine translation. In *Proceedings of the EACL 2017 Software Demonstrations*. Association for Computational Linguistics, pages 1-4.

Swasti Singhal, and Monika Jena. 2013. A study on WEKA tool for data preprocessing, classification and clustering. *International Journal of Innovative Technology and Exploring Engineering*, 2(6): 250-253.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas*. The Association for Machine Translation in the Americas, pages 223-231.

Vilelmini Sosoni, Katia Lida Kermanidis, Maria Stasimioti, Thanasis Naskos, Eirini Takoulidou, Men-

no van Zaanen, Sheila Castilho, Panayota Georgakopoulou, Valia Kordoni, and Markus Egg. 2018. Translation Crowdsourcing: Creating a Multilingual Corpus of Online Educational Content. In *Proceedings of the 11th International Conference on Language Resources and Evaluation.* European Language Resources Association, pages 479-483.

Ilya Sutskever, Vinyals Oriol, and Le V. Quoc. 2014. Sequence to sequence learning with neural networks. *In Proceedings of the Advances in Neural Information Processing Systems*. NIPS, pages 8–13.

Klaus Zechner, and Alex Waibel. 2000. Minimizing word error rate in textual summaries of spoken language. *1st Meeting of the North American Chapter of the Association for Computational Linguistics*.

Appendix

ID	Src	T1	T2	Tr
1	And so you end up with, you know, hoods and flagella and little nodules on the end of stems and what have you.	Και έτσι καταλήγεις, ξέρετε, κουκούλες και flagella και η μικρή οζίδια στο τέλος του κοτσάνια και τι έχετε.	Και έτσι καταλήγετε με, ξέρετε, κουκίδες και πλάκες μαστιγίων και μικρές ακίδες στο τέλος των μίσχων και σε αυτό που έχετε.	Και έτσι καταλήγετε με , ξέρετε , κουκούλες και βλεφαρίδες και μικρούς θρόμβους στο τέλος των βλαστών και διάφορα άλλα τέτοια.
2	In Discipline and Punish, Michel Foucault described the Panopticon as the "perfect" instrument of power.	Στο Επιτήρηση και Τιμωρία, Michel Foucault περιέγραψε την Panopticon ως το "τέλειο" όργανο εξουσίας.	Στην πειθαρχι ? α και στο Πω ? νητο, ο Μισέλ Φουκούλτ περιέγραψε το Πανοπικόν ως το "τέλειο" όργανο δύναμης.	Στο έργο του Επιτήρηση και Τιμωρία, ο Michel Foucault περιέγραψε το Πανοπτικό ως το «τέλειο» όργανο εξουσίας.
3	That's why I would like you to start taking sumo lessons, because... just look at those delicious guys! it's obvious that sumo fighting gets you fat!"	Γι' αυτό θα ήθελα να αρχίσουμε να παίρνουμε μαθήματα σούμο, επειδή... απλά κοιτάξτε αυτά τα υπέροχα παιδιά! Είναι προφανές ότι σούμο μάχες παίρνει το λίπος!	Γι' αυτό θα ήθελα να αρχίσετε να παίρνετε μαθήματα αθροίσματος, γιατί... κοιτάξτε αυτούς τους γευστικότερους τύπους! Είναι προφανές ότι οι μάχες στο sumo σας παχαίνουν!	Γι' αυτό θα ήθελα να αρχίσεις μαθήματα σούμο, γιατί... κοίτα αυτούς τους νόστιμους τύπους! Είναι προφανές ότι το σούμο σε παχαίνει!

Table 3: Linguistic Analysis