Estimating Grammar Correctness for a Priori Estimation of Machine Translation Post-Editing Effort

Nicholas H. Kirk, Guchun Zhang
Alpha Calligraphic Research Cambridge Ltd.
St Andrew's House, St Andrew's Road,
Cambridge CB4 1DL, UK
{nkirk,gzhang}@alphacrc.com

Georg Groh
Fakultat für Informatik
Technische Universität München,
Germany
grohg@in.tum.de

Abstract

We present a supervised learning pilot application for estimating Machine Translation (MT) output reusability, in view of supporting a human post-editor of MT content. We train our model on typed dependencies (labeled grammar relationships) extracted from human reference and raw MT data, to then predict grammar relationship correctness values that we aggregate to provide a binary segment-level evaluation. In view of scaling up to larger data, we provide implemented Naïve Bayes and Stochastic Gradient Descent with Support Vector Machine loss function approaches and their evaluation, and verify the correlation of predicted values with human judgement.

1 Introduction

Currently the Machine Translation (MT) research community attempts to seamlessly integrate both humans and MT-instances in the workflow of textual translation. Efforts towards this integration focus, for instance, on automating a posteriori processes such as post-editing (Simard et al., 2007), or other format coherence maintenance (e.g. date, spelling). Our contribution addresses cases when a post-editor has to start a segment from scratch, because the MT raw output turns out to be a hindrance rather than an aid, and the corresponding evaluation time between editing and manually retranslating a sequence is wasted a posteriori. Reasons for unusable MT output in this context could potentially be a combination of the following or more factors, with reference to the target segment:

- the word order, or grammar, are such that the sentence structure is unintelligible
- the lexical semantics of the words do not convey the meaning of the source segment

These lexical or structural factors can be present to various extents and their threshold of identification can be subjective for each post-editor, but hypothetically any intervention on the latter points is quantifiable in terms of post-editing time, being this the most observable aspect of post-editing effort (Krings, 2001). This paper proposes a supervised learning approach to discriminate typed grammar relation instances that compose a human-written sentence from any other form, in order to identify segments that can potentially lead to time loss on the basis of its incorrect grammar or word adjacency and delete them before post-editing. The remainder presents the project's assumptions and the nature of the adopted learning features (Section 2), the high-level algorithmic approach and the theory behind the adopted prediction models (Section 3). We then provide our implementation outline and the evaluation approaches (Section 4). In conclusion we present current limitations (Section 5), related and future work (Section 6), and the conclusions (Section 7).

2 Concept

We will now provide some background and a series of assertions as regards creating a classification method to estimate MT output grammar correctness, which mainly aims to support the post-editor in assessing which segments will take longer to post-edit than to translate from scratch. We assume the following post-editing behavioral phases:

1. Read source and/or target to various extents, in order to:
   - check grammatical consistency of target
   - check whether semantics have been conveyed between source and target
2. Insert or delete text accordingly

Workshop on Humans and Computer-assisted Translation, pages 16–21,
Gothenburg, Sweden, 26 April 2014.

Given the lack of robust adequacy understanding methods (i.e. verifying meaning conveyance), we will perform analysis at a grammar and word order level, and for this we will seek a grammar-related formalism that is informative, scalable and robust to multiple, potentially unseen grammar instance variants. We therefore exploit typed dependencies (De Marneffe and Manning, 2008), a labeled, directed grammar relationship among pairs of words, which provides information on the order of arguments and their relationship type. Figure 1 shows words of a segment instance, and for each of these the unary and binary predicates of Part-Of-Speech tagging and typed dependency, respectively.

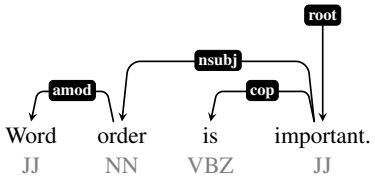

Figure 1: Example of words that compose a segment instance, their typed dependencies (illustrated as labeled directed edges), and the Part-Of-Speech (POS) tags.

3 Algorithm

Having discussed our aim and the required informativeness, we present a pipeline for both training a hypothesis model and the prediction itself (Figure 2). The process comprises a typed dependency extraction module (M1) that, given a set of sentences from a `test` or `training` text, provides instances of grammar relationships in the form of two word arguments ($arg1$, $arg2$) and a type label ($deptype$), which we will adopt as features. In the training phase, a training module (M2) labels the feature data obtained from $\{trainHuman\}$ instances as 1, and from $\{trainHuman \setminus trainMT\}$ as 0, where $\setminus$ is the set difference operator. We therefore consider any typed dependency instance that does not appear in the human reference text as 'bad'. This assumption holds when training on large datasets that comprise different grammar variants. From such labeled dataset, M2 then formulates a hypothesis model. More details on generating the hypothesis are provided in the next paragraph. During a test phase, various instances of a prediction module (M3) exploit such hypothesis model

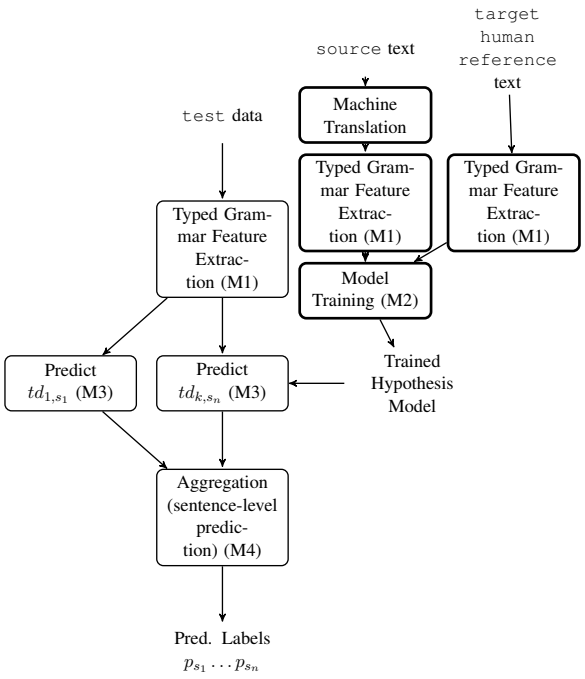

Figure 2: Abstract implementation pipeline for training the hypothesis model (in bold), and for the prediction itself of a typed dependency-based segment reusability estimator.

to predict the grammar relationship goodness values of $td_{1,s_i} \dots td_{k,s_i}$ for a sentence s_i and its typed dependencies $td_1 \dots td_k$, for all sentences $s_i \in \{s_1 \dots s_n\}$. A final phase (M4) aggregates the output predictions of grammar relationships for each sentence, in order to construct segment-level estimations (Equation 1).

$$P_{s_i} = \frac{1}{k} \sum_{j=1}^{k} P_{td_{j,s_i}} \quad (1)$$

Hypothesis Model Given our aim to achieve method robustness and breadth of applicability, and that context abstraction is potentially achievable since correctness of a grammar relationship is not dependent on neighbouring dependencies, we train on a high number of diverse training instances. In order to obtain a reduction in time and space complexity, we approximate our hypothesis by making sample independence assumptions, such as the Naïve Bayes (NB) approach (Good, 1965). NB is a model that assumes feature independence, i.e. the 'naiveness' implies that every feature F_i is conditionally independent of every other feature F_j for $j \neq i$ given the class C. Equation 2 describes the core of all current NB variants.

$$p\left(C|F_1 \dots F_n\right) \propto p\left(C\right) \prod_{i=1}^{n} p\left(F_i|C\right) \quad (2)$$

Such generative model is efficient and requires just one linear iteration for training, hence its suitability for large input scaling. Unfortunately, the modeling assumptions that enable efficient computability come at the expense of accuracy. More precisely, NB-based methods maximize likelihood conditioned only over the class label C, and not over the set of all other remaining features, and as a result its effectiveness is often outperformed by discriminative classifiers such as Support Vector Machines (SVM) (Crammer and Singer, 2002; Puurula, 2012). However, given its high efficiency and scalability, we will use NB as our primary model and compare it with an algorithm that has deeper modeling assumptions, but exploits inference approximation to reduce complexity, namely SVM with Stochastic Gradient Descent for parameter finding. Approximated inference is often achieved via traditional gradient descent methods (see Equation 3 for linear classification or regression approaches) that are largely used, first-order, stepwise optimization algorithms that seek minima of a problem with large dimensionality and unknown convexity status.

$$w_{t+1} = w_t - \gamma_t \frac{1}{n} \sum_{i=1}^{n} \nabla_w Q(z_t, w_t) \quad (3)$$

where our objective is to iteratively minimize, given an initial parameterization (starting point w_0, number of iterations, step length γ_0), and a function $Q(w)$, or more specifically when within the machine learning context, an *empirical risk* $E_n(f)$, defined as:

$$Q(w) = E_n(f) = \frac{1}{n} \sum_{i=1}^{n} l\left(f\left(x_i\right), y_i\right) \quad (4)$$

where in turn $l(\hat{y}, y)$ is a *loss function* that quantifies the cost of predicting $\hat{y}$ when the actual answer is y. One advantage is computational efficiency, while the disadvantages include the inability to provide certainty of termination or result determinism. Recent literature partially circumvents these problems (Hager and Zhang, 2005), and the use of such family of algorithms has been rediscovered for large scale data learning, whose applications prefer approximate over exact inference, by using a stochastic variant (Equation 5) of the traditional method (Zhang, 2004; Bottou, 2010), which samples a subset of the training data.

$$w_{t+1} = w_t - \gamma_t \nabla_w Q(z_t, w_t) \quad (5)$$

We adopt a *hinge loss* function for Support Vector Machine (SVM) classification, a method which has proven to be reliable for elaborating high scale, sparsely distributed instance vector applications that have dense concept vectors (Joachims, 1998; Rosasco et al., 2004).

4 Implementation & Evaluation

As a proof-of-concept and means of evaluation, we constructed a Java prototype that implements the pipeline in Figure 2, by making use of the Moses process (Koehn et al., 2007) for machine translation, the Stanford Parser (De Marneffe and Manning, 2008) for typed dependency extraction, the API of the general-purpose machine learning analysis platform Weka (Hall et al., 2009) for training and prediction, and ad-hoc implementations of the remaining specified modules. Typed dependency extraction is possible by feeding a pre-trained language-specific Probabilistic Context-Free Grammar (PCFG) into the parser, which is a model that defines probabilistic grammar production rules for such language. Such a model is trained beforehand on a large, syntactically annotated text corpus (i.e. a treebank) (Jelinek et al., 1992).

4.1 Experiment

We used our prototype on a subset of the Europarl test data (Koehn, 2005), extracting 536404 instances from 11270 human reference lines, and from 11270 machine-translated lines that share the same source. Our Naive Bayes algorithm (using word frequencies, no pruning) required 1.08 seconds to formulate a hypothesis model, versus the 28613.64 seconds required for the construction of a Stochastic Gradient Descent model (hinge loss function for SVM, step length 0.01, 500 steps, no pruning). A 10-fold cross validation on the training set provided a correct instance classification of 67.0168% for NB model, versus the 66.0118% of the SGD model. Table 1 provides further statistics of the latter evaluations.

		Precision	Recall	F-1
NB	'good'	0.665	0.839	0.742
	'bad'	0.683	0.451	0.543
SGD	'good'	0.646	0.883	0.746
	'bad'	0.709	0.370	0.487

Table 1: Precision and recall values of the 10-fold cross validation for both NB and SGD methods

4.2 Correlation with Human Judgement

We organized a survey among post-editors to gather human judgement values on a small set of 80 segments, for independent analysis and for comparison with machine predicted labels. Four translators with different experience level (in terms of years, 10+,5,1+, 1) evaluated a questionnaire of 80 Europarl-domain segments. Half of these were official human reference Europarl segments, while the other half were MT processed $FR \rightarrow EN$ segments. The process first involved a binary labeling task $\{\texttt{aid|hindrance}\}$ (two instances of results are in Figure 3), and the second a phase assigning a mark $\{\texttt{0...10}\}$ to define its level of usefulness. Together with the $lineNumber$, we will consider these three features and their data as the human judgement dataset. By aggregating the human binary evaluations with a majority vote and comparing these with the sentence-level prediction of our system, *NB correctly classified 82.5% of the segments, while SGD classified 83.75%*.

Preliminary clustering analysis on the latter confirmed the intuitive idea of the subjectivity of this kind of reusability evaluation for each translator. An unsupervised categorization was performed using an Expectation Maximization (EM) of Gaussian mixtures on the human judgement survey dataset, to understand distributional properties and use this as a basis to evaluate how the prototype results correlate with human judgement. Starting with a human-only dataset and no initial prior, the EM algorithm estimated by cross-validation only one homogeneous cluster. By then providing the number of clusters (i.e. the number of human translators, 4), the cluster evaluation assignments were not aligned with the number of instances present for each translator, which implies post-editor behavior indistinguishability via this method. Once machine predicted samples are added to the evaluation set, a further step is to verify whether the cluster assignments are within an acceptable neighborhood of the previous cluster assignment values. As shown in Table 2, cluster assignment percentages of NB predicted labels are closer to the original cluster assignments from the training set than SGD value-based cluster assignments. The clustering approach described will be used as a preliminary method for effectiveness evaluation, i.e. by evaluating the extent to which machine predicted values are a mixture of human behavior data based on the cluster assignment value distance from the generating model values.

label	eval.: human		eval.: NB		eval.: SGD	
0	179	(56%)	52	(65%)	58	(73%)
1	24	(8%)	3	(4%)	2	(3%)
2	52	(16%)	7	(9%)	2	(3%)
3	65	(20%)	18	(23%)	18	(23%)

Table 2: Evaluation data obtained with the cluster model generated from the human judgement dataset, with the number of clusters defined as 4.

```
1) on the issue of Jerusalem , they
have shown in a spirit of openness
and a capacity for listening hopeless .

2) that is completely disproportionate
and it does no favours for the
peace process .
```

Figure 3: Examples of segments classified as 'bad' (1) and 'good' (2) by all the post-editors of the experiment described in Section 4.2.

5 Limitations

Method robustness would imply that the grammar relationships under test are known, or that the prediction algorithm reacts well to unseen data. Figure 4 presents an example that shows how typed dependencies of two related sentences (namely reference and MT output of the same source) and the word usage itself can be scarcely related, or not overlap. This highlights that we cannot assume training coverage of the typed dependencies in the test segment, even if the contained words are present in the training set in multiple grammatical contexts. This stresses the importance of the scaling requirement and the complexity reduction measures stated in Section 3, in order to train diverse grammatical instance variants. A further

aspect to consider with the Naïve Bayes formulation is that the model defines a likelihood for each entry conditioned on an unconditional class probability, which is correlated to the ratio of 'bad' and 'good' grammar relationships present in the training set. This information usage decreases robustness, as the model captures quality information of the MT instance, which can be subject to variability (e.g. the language pair, MT instance setup).

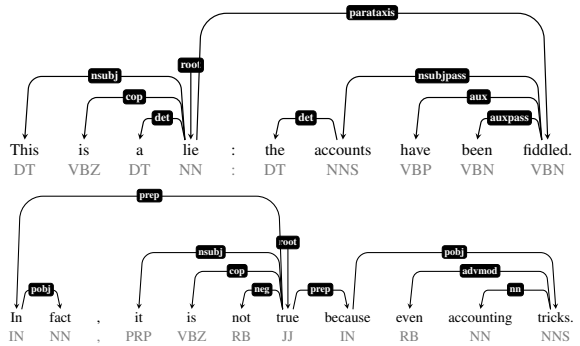

Figure 4: Human reference and raw MT output derived from the test subset of Europarl FR-EN corpus, which show typed dependency relationships, the words, and related part-of-speech tags that highlight possible word usage variance

6 Related & Future Work

In order to estimate MT output quality, literature in the past has traditionally compared an automatically translated sentence to one or more human text references (Papineni et al., 2002), while other work has exploited unlabeled dependencies in order to take into account legitimate grammatical or lexical choice variations (Liu and Gildea, 2005). Other work improves the classification effectiveness of the latter by considering typed dependencies (Owczarzak et al., 2007). Some data-driven, referenceless evaluation approaches to learning human judgement have been introduced (Corston-Oliver et al., 2001), which exploit syntactic features and linguistic indicators (Gamon et al., 2005), but have also been combined with typed dependency features (He and Way, 2009). Estimation of post-editing effort is a growing concern addressed by Confidence Estimation (CE) (Specia, 2011), but so far, to the best of our knowledge, work within the domain performs supervised learning of statistical linguistic features (Felice and Specia, 2012), but not of dependency features, i.e. the main focus of this contribution. Previous quality estimation methods differ in nature from the presented view, given that they attempt to predict a discrete level of post-editing effort (Bojar et al., 2013), more subject to annotation subjectivity, or to perform binary classification (Hardmeier, 2011), but do not focus on segment reusability estimation. Future work will focus on testing the hypothesis modeling and feature extraction for scalability on larger context-abstract data, and verifying the distinguishability of predicted values from more human-annotated judgements using the method stated in Section 4.2. Time gain values have not yet been acquired given the unusability of the time productivity metrics currently favored, which do not exhibit direct correlation with real PE time, and are also focus of future investigations. Futhermore, evaluation on the WMT Quality Estimation Shared Task datasets will be performed, for comparisons with state of the art methods of post-editing effort quantification (Bojar et al., 2013).

7 Conclusions

The presented pilot study proposes a grammar-based analysis for categorizing MT output in terms of whether it is an aid or a hindrance to the post-editor. Our contributions are mainly the (i) use of typed dependency learning for binary evaluation of confidence estimation and (ii) the analysis of adequate algorithmic solutions to achieve its scalability and context abstraction. Preliminary results show that aggregation of predictions operated at a typed dependency level provide an evaluation that resembles the segment-level judgement displayed by post-editors. Futhermore, for the hypothesis models created on the dataset tested, Naïve Bayes outperformed Stochastic Gradient Descent with hinge loss for Support Vector Machine in terms of training efficiency, and is on a par regarding classification effectiveness. We have showed the preliminary advantages of typed dependency-based estimation in terms of context abstraction, which provides a novel type of assistance to human post-editors and correlates with post-editing cost rather than commonly analyzed linguistic metrics.

References

Ondřej Bojar, Christian Buck, Chris Callison-Burch, Christian Federmann, Barry Haddow, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2013. Findings of the 2013 Work-

shop on Statistical Machine Translation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 1–44, Sofia, Bulgaria, August. Association for Computational Linguistics.

Léon Bottou. 2010. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer.

Simon Corston-Oliver, Michael Gamon, and Chris Brockett. 2001. A machine learning approach to the automatic evaluation of machine translation. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 148–155. Association for Computational Linguistics.

Koby Crammer and Yoram Singer. 2002. On the learnability and design of output codes for multiclass problems. *Machine Learning*, 47(2-3):201–233.

Marie-Catherine De Marneffe and Christopher D Manning. 2008. The stanford typed dependencies representation. In *Coling 2008: Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation*, pages 1–8. Association for Computational Linguistics.

Mariano Felice and Lucia Specia. 2012. Linguistic features for quality estimation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 96–103. Association for Computational Linguistics.

Michael Gamon, Anthony Aue, and Martine Smets. 2005. Sentence-level mt evaluation without reference translations: Beyond language modeling. In *Proceedings of EAMT*, pages 103–111.

Irving John Good. 1965. *The estimation of probabilities: An essay on modern Bayesian methods*, volume 30. MIT press Cambridge, MA.

William W Hager and Hongchao Zhang. 2005. A new conjugate gradient method with guaranteed descent and an efficient line search. *SIAM Journal on Optimization*, 16(1):170–192.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. 2009. The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18.

Christian Hardmeier. 2011. Improving machine translation quality prediction with syntactic tree kernels. In *Proceedings of the 15th conference of the European Association for Machine Translation (EAMT 2011)*, pages 233–240.

Yifan He and Andy Way. 2009. Learning labelled dependencies in machine translation evaluation.

Frederick Jelinek, John D Lafferty, and Robert L Mercer. 1992. *Basic methods of probabilistic context free grammars*. Springer.

Thorsten Joachims. 1998. *Text categorization with support vector machines: Learning with many relevant features*. Springer.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 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*, pages 177–180. Association for Computational Linguistics.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5.

Hans P Krings. 2001. *Repairing texts: empirical investigations of machine translation post-editing processes*, volume 5. Kent State University Press.

Ding Liu and Daniel Gildea. 2005. Syntactic features for evaluation of machine translation. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 25–32.

Karolina Owczarzak, Josef Van Genabith, and Andy Way. 2007. Dependency-based automatic evaluation for machine translation. In *Proceedings of the NAACL-HLT 2007/AMTA Workshop on Syntax and Structure in Statistical Translation*, pages 80–87. Association for Computational Linguistics.

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*, pages 311–318. Association for Computational Linguistics.

Antti Puurula. 2012. Combining modifications to multinomial naive bayes for text classification. In *Information Retrieval Technology*, pages 114–125. Springer.

Lorenzo Rosasco, Ernesto De Vito, Andrea Caponnetto, Michele Piana, and Alessandro Verri. 2004. Are loss functions all the same? *Neural Computation*, 16(5):1063–1076.

Michel Simard, Cyril Goutte, and Pierre Isabelle. 2007. Statistical phrase-based post-editing.

Lucia Specia. 2011. Exploiting objective annotations for measuring translation post-editing effort. In *Proceedings of the 15th Conference of the European Association for Machine Translation*, pages 73–80.

Tong Zhang. 2004. Solving large scale linear prediction problems using stochastic gradient descent algorithms. In *Proceedings of the twenty-first international conference on Machine learning*, page 116. ACM.