Learning to Differentiate Better from Worse Translations

Francisco Guzmán  Shafiq Joty  Luís Márquez
Alessandro Moschitti  Preslav Nakov  Massimo Nicosia
ALT Research Group
Qatar Computing Research Institute — Qatar Foundation
{fguzman,sjoty,lmarquez,amoschitti,pnakov,mnicosia}@qf.org.qa

Abstract

We present a pairwise learning-to-rank approach to machine translation evaluation that learns to differentiate better from worse translations in the context of a given reference. We integrate several layers of linguistic information encapsulated in tree-based structures, making use of both the reference and the system output simultaneously, thus bringing our ranking closer to how humans evaluate translations. Most importantly, instead of deciding upfront which types of features are important, we use the learning framework of preference re-ranking kernels to learn the features automatically. The evaluation results show that learning in the proposed framework yields better correlation with humans than computing the direct similarity over the same type of structures. Also, we show our structural kernel learning (SKL) can be a general framework for MT evaluation, in which syntactic and semantic information can be naturally incorporated.

1 Introduction

As a result, this has enabled rapid development in the field of statistical machine translation (SMT), by allowing to train and tune systems as well as to track progress in a way that highly correlates with human judgments.

Today, MT evaluation is an active field of research, and modern metrics perform analysis at various levels, e.g., lexical (Papineni et al., 2002; Snover et al., 2006), including synonymy and paraphrasing (Lavie and Denkowski, 2009); syntactic (Giménez and Márquez, 2007; Popović and Ney, 2007; Liu and Gildea, 2005); semantic (Giménez and Márquez, 2007; Lo et al., 2012); and discourse (Comelles et al., 2010; Wong and Kit, 2012; Guzmán et al., 2014; Joty et al., 2014).

Automatic MT evaluation metrics compare the output of a system to one or more human references in order to produce a similarity score. The quality of such a metric is typically judged in terms of correlation of the scores it produces with scores given by human judges. As a result, some evaluation metrics have been trained to reproduce the scores assigned by humans as closely as possible (Albrecht and Hwa, 2008). Unfortunately, humans have a hard time assigning an absolute score to a translation. Hence, direct human evaluation scores such as adequacy and fluency, which were widely used in the past, are now discontinued in favor of ranking-based evaluations, where judges are asked to rank the output of 2 to 5 systems instead. It has been shown that using such ranking-based assessments yields much higher inter-annotator agreement (Callison-Burch et al., 2007).

While evaluation metrics still produce numerical scores, in part because MT evaluation shared tasks at NIST and WMT ask for it, there has also been work on a ranking formulation of the MT evaluation task for a given set of outputs. This was shown to yield higher correlation with human judgments (Duh, 2008; Song and Cohn, 2011).
Learning automatic metrics in a pairwise setting, i.e., learning to distinguish between two alternative translations and to decide which of the two is better (which is arguably one of the easiest ways to produce a ranking), emulates closely how human judges perform evaluation assessments in reality. Instead of learning a similarity function between a translation and the reference, they learn how to differentiate a better from a worse translation given a corresponding reference. While the pairwise setting does not provide an absolute quality scoring metric, it is useful for most evaluation and MT development scenarios.

In this paper, we propose a pairwise learning setting similar to that of Duh (2008), but we extend it to a new level, both in terms of feature representation and learning framework. First, we integrate several layers of linguistic information encapsulated in tree-based structures; Duh (2008) only used lexical and POS matches as features. Second, we use information about both the reference and two alternative translations simultaneously, thus bringing our ranking closer to how humans rank translations. Finally, instead of deciding upfront which types of features between hypotheses and references are important, we use a our structural kernel learning (SKL) framework to generate and select them automatically.

The structural kernel learning (SKL) framework we propose consists in: (i) designing a structural representation, e.g., using syntactic and discourse trees of translation hypotheses and a reference; and (ii) applying structural kernels (Moschitti et al., 2007, Moschitti, 2008), to such representations in order to automatically inject structural features in the preference re-ranking algorithm. We use this method with translation-reference pairs to directly learn the features themselves, instead of learning the importance of a predetermined set of features. A similar learning framework has been proven to be effective for question answering (Moschitti et al., 2007), and textual entailment recognition (Zanzotto and Moschitti, 2006).

Our goals are twofold: (i) in the short term, to demonstrate that structural kernel learning is suitable for this task, and can effectively learn to rank hypotheses at the segment-level; and (ii) in the long term, to show that this approach provides a unified framework that allows to integrate several layers of linguistic analysis and information and to improve over the state-of-the-art.

Below we report the results of some initial experiments using syntactic and discourse structures. We show that learning in the proposed framework yields better correlation with humans than applying the traditional translation–reference similarity metrics using the same type of structures. We also show that the contributions of syntax and discourse information are cumulative. Finally, despite the limited information we use, we achieve correlation at the segment level that outperforms BLEU and other metrics at WMT12, e.g., our metric would have been ranked higher in terms of correlation with human judgments compared to TER, NIST, and BLEU in the WMT12 Metrics shared task (Callison-Burch et al., 2012).

2 Kernel-based Learning from Linguistic Structures

In our pairwise setting, each sentence $s$ in the source language is represented by a tuple $(t_1, t_2, r)$, where $t_1$ and $t_2$ are two alternative translations and $r$ is a reference translation. Our goal is to develop a classifier of such tuples that decides whether $t_1$ is a better translation than $t_2$ given the reference $r$.

Engineering features for deciding whether $t_1$ is a better translation than $t_2$ is a difficult task. Thus, we rely on the automatic feature extraction enabled by the SKL framework, and our task is reduced to choosing: (i) a meaningful structural representation for $(t_1, t_2, r)$, and (ii) a feature function $\phi_{mt}$ that maps such structures to substructures, i.e., our feature space. Since the design of $\phi_{mt}$ is complex, we use tree kernels applied to two simpler structural mappings $\phi_M(t_1, r)$ and $\phi_M(t_2, r)$. The latter generate the tree representations for the translation-reference pairs $(t_1, r)$ and $(t_2, r)$. The next section shows such mappings.

2.1 Representations

To represent a translation-reference pair $(t, r)$, we adopt shallow syntactic trees combined with RST-style discourse trees. Shallow trees have been successfully used for question answering (Severyn and Moschitti, 2012) and semantic textual similarity (Severyn et al., 2013b); while discourse information has proved useful in MT evaluation (Guzmán et al., 2014; Joty et al., 2014). Combined shallow syntax and discourse trees worked well for concept segmentation and labeling (Saleh et al., 2014a).
Figure 1 shows two example trees combining discourse, shallow syntax and POS: one for a translation hypothesis (top) and the other one for the reference (bottom). To build such structures, we used the Stanford POS tagger (Toutanova et al., 2003), the Illinois chunker (Punyakanok and Roth, 2001), and the discourse parser of (Joty et al., 2012; Joty et al., 2013).

The lexical items constitute the leaves of the tree. The words are connected to their respective POS tags, which are in turn grouped into chunks. Then, the chunks are grouped into elementary discourse units (EDU), to which the nuclearity status is attached (i.e., NUCLEUS or SATELLITE). Finally, EDUs and higher-order discourse units are connected by discourse relations (e.g., DIS:ELABORATION).

2.2 Kernels-based modeling

In the SKL framework, the learning objects are pairs of translations \( (t_1, t_2) \). Our objective is to automatically learn which pair features are important, independently of the source sentence. We achieve this by using kernel machines (KMs) over two learning objects \( (t_1, t_2), (t_1', t_2') \), along with an explicit and structural representation of the pairs (see Fig. 1).

More specifically, KMs carry out learning using the scalar product

\[
K_{mt}(t_1, t_2, \langle t_1', t_2' \rangle) = \phi_{mt}(t_1, t_2) \cdot \phi_{mt}(t_1', t_2')
\]

where \( \phi_{mt} \) maps pairs into the feature space.

Considering that our task is to decide whether \( t_1 \) is better than \( t_2 \), we can conveniently represent the vector for the pair in terms of the difference between the two translation vectors, i.e., \( \phi_{mt}(t_1, t_2) = \phi_K(t_1) - \phi_K(t_2) \). We can approximate \( K_{mt} \) with a preference kernel \( PK \) to compute this difference in the kernel space \( K \):

\[
PK(t_1, t_2, \langle t_1', t_2' \rangle)
= K(t_1) - \phi_K(t_2) \cdot (\phi_K(t_1') - \phi_K(t_2'))
= K(t_1, t_1') + K(t_2, t_2') - K(t_1, t_2') - K(t_2, t_1')
\]

The advantage of this is that now \( K(t_1, t_2) = \phi_K(t_1) \cdot \phi_K(t_2) \) is defined between two translations only, and not between two pairs of translations. This simplification enables us to map translations into simple trees, e.g., those in Figure 1, and then to apply them tree kernels, e.g., the Partial Tree Kernel (Moschitti, 2006), which carry out a scalar product in the subtree space.

We can further enrich the representation \( \phi_K \), if we consider all the information available to the human judges, when they are ranking translations. That is, the two alternative translations along with their corresponding reference.

---

\(^1\)The discourse parser can be downloaded from [http://alt.qcri.org/tools/](http://alt.qcri.org/tools/)
In particular, let \( r \) and \( r' \) be the references for the pairs \((t_1, t_2)\) and \((t'_1, t'_2)\), we can redefine all the members of Eq. 1, e.g., \( K(t_1, t'_1) \) becomes
\[
K((t_1, r), (t'_1, r')) = PTK(\phi_M(t_1, r), \phi_M(t'_1, r')) + PTK(\phi_M(r, t_1), \phi_M(r', t'_1))
\]
where \( \phi_M \) maps a pair of texts to a single tree.

There are several options to produce the bitext-to-tree mapping for \( \phi_M \). A simple approach is to only use the tree corresponding to the first argument of \( \phi_M \). This leads to the basic model
\[
K((t_1, r), (t'_1, r')) = PTK(\phi_M(t_1), \phi_M(t'_1)) + PTK(\phi_M(r), \phi_M(r'))\]
where \( \phi_M \) can be defined as sum of two tree kernels applied to the trees constructed by \( \phi_M \) (we previously informally mentioned it).

However, this simple mapping may be ineffective since the trees within a pair, e.g., \((t_1, r)\), are treated independently, and no meaningful features connecting \( t_1 \) and \( r \) can be derived from their tree fragments. Therefore, we model \( \phi_M(r, t_1) \) by using word-matching relations between \( t_1 \) and \( r \), such that connections between words and constituents of the two trees are established using position-independent word matching. For example, in Figure 1, the thin dashed arrows show the links connecting the matching words between \( t_1 \) and \( r \). The propagation of these relations works from the bottom up. Thus, if all children in a constituent have a link, their parent is also linked.

The use of such connections is essential as it enables the comparison of the structural properties and relations between two translation-reference pairs. For example, the tree fragment \([\text{ELABORATION} [\text{SATELLITE}]]\) from the translation is connected to \([\text{ELABORATION} [\text{SATELLITE}]]\) in the reference, indicating a link between two entire discourse units (drawn with a thicker arrow), and providing some reliability to the translation.

Note that the use of connections yields a graph representation instead of a tree. This is problematic as effective models for graph kernels, which would be a natural fit to this problem, are not currently available for exploiting linguistic information. Thus, we simply use \( K \), as defined above, where the mapping \( \phi_M(t_1, r) \) only produces a tree for \( t_1 \) annotated with the marker REL representing the connections to \( r \). This marker is placed on all node labels of the tree generated from \( t_1 \) that match labels from the tree generated from \( r \).

In other words, we only consider the trees enriched by markers separately, and ignore the edges connecting both trees.

3 Experiments and Discussion

We experimented with datasets of segment-level human rankings of system outputs from the WMT11 and the WMT12 Metrics shared tasks (Callison-Burch et al., 2011; Callison-Burch et al., 2012): we used the WMT11 dataset for training and the WMT12 dataset for testing. We focused on translating into English only, for which the datasets can be split by source language: Czech (cs), German (de), Spanish (es), and French (fr). There were about 10,000 non-tied human judgments per language pair per dataset. We scored our pairwise system predictions with respect to the WMT12 human judgments using the Kendall’s Tau (\( \tau \)), which was official at WMT12.

Table 1 presents the \( \tau \) scores for all metric variants introduced in this paper: for the individual language pairs and overall. The left-hand side of the table shows the results when using as similarity the direct kernel calculation between the corresponding structures of the candidate translation and the reference, e.g., as in (Guzmán et al., 2014; Joty et al., 2014). The right-hand side contains the results for structured kernel learning.

We can make the following observations:

(i) The overall results for all SKL-trained metrics are higher than the ones when applying direct similarity, showing that learning tree structures is better than just calculating similarity.

(ii) Regarding the linguistic representation, we see that, when learning tree structures, syntactic and discourse-based trees yield similar improvements with a slight advantage for the former. More interestingly, when both structures are put together in a combined tree, the improvement is cumulative and yields the best results by a sizable margin. This provides positive evidence towards our goal of a unified tree-based representation with multiple layers of linguistic information.

(iii) Comparing to the best evaluation metrics that participated in the WMT12 Metrics shared task, we find that our approach is competitive and would have been ranked among the top 3 participants.

\[^2\text{Note that a non-pairwise model, i.e., } K(t_1, r), \text{ could also be used to match the structural information above, but it would not learn to compare it to a second pair } (t_2, r).\]

\[^3\text{Applying tree kernels between the members of a pair to generate one feature (for each different kernel function) has become a standard practice in text similarity tasks (Severyn et al., 2013b) and in question answering (Severyn et al., 2013a).}\]
Furthermore, our result (0.237) is ahead of the correlation obtained by popular metrics such as TER (0.217), NIST (0.214) and BLEU (0.185) at WMT12. This is very encouraging and shows the potential of our new proposal.

In this paper, we have presented only the first exploratory results. Our approach can be easily extended with richer linguistic structures and further combined with some of the already existing strong evaluation metrics.

Note that the results in Table 1 were for training on WMT11 and testing on WMT12 for each language pair in isolation. Next, we study the impact of the choice of training language pair. Table 2 shows cross-language evaluation results for DIS+SYN: lines 1-4 show results when training on WMT11 for one language pair, and then testing for each language pair of WMT12.

We can see that the overall differences in performance (see the last column: all) when training on different source languages are rather small, ranging from 0.209 to 0.221, which suggests that our approach is quite independent of the source language used for training. Still, looking at individual test languages, we can see that for de-en and es-en, it is best to train on the same language; this also holds for fr-en, but there it is equally good to train on es-en. Interestingly, training on es-en improves a bit for cs-en.

These somewhat mixed results have motivated us to try tuning on the full WMT11 dataset; as line 5 shows, this yielded improvements for all language pairs except for es-en. Comparing to line 4 in Table 1, we see that the overall Tau improved from 0.231 to 0.237.

### 4 Conclusions and Future Work

We have presented a pairwise learning-to-rank approach to MT evaluation, which learns to differentiate good from bad translations in the context of a given reference. We have integrated several layers of linguistic information (lexical, syntactic and discourse) in tree-based structures, and we have used the structured kernel learning to identify relevant features and learn pairwise rankers.

The evaluation results have shown that learning in the proposed SKL framework is possible, yielding better correlation (Kendall’s τ) with human judgments than computing the direct kernel similarity between translation and reference, over the same type of structures. We have also shown that the contributions of syntax and discourse information are cumulative, indicating that this learning framework can be appropriate for the combination of different sources of information. Finally, despite the limited information we used, we achieved better correlation at the segment level than BLEU and other metrics in the WMT12 Metrics task.

In the future, we plan to work towards our long-term goal, i.e., including more linguistic information in the SKL framework and showing that this can help. This would also include more semantic information, e.g., in the form of Brown clusters or using semantic similarity between the words composing the structure calculated with latent semantic analysis (Saleh et al., 2014b).

We further want to show that the proposed framework is flexible and can include information in the form of quality scores predicted by other evaluation metrics, for which a vector of features would be combined with the structured kernel.

### Acknowledgments

This research is part of the Interactive sYstems for Answer Search (Iyas) project, conducted by the Arabic Language Technologies (ALT) group at Qatar Computing Research Institute (QCRI) within the Qatar Foundation.
References


