The Goal
Implement the core Translation Memory (TM) functionality: given a source-language sentence, find the best matching translation in a bilingual corpus. Our goal is to build a reasonable research prototype, focusing on output quality rather than time-space efficiency.

Motivations
- Much recent work on post-editing and computer-assisted translation (CAT) ignores the fact that most translators are already using CAT tools, the tools most common of which are TM systems. In the short term, new tools are not going to supplant TMs, but rather complement them.
- The problem for researchers who wish to experiment with TM technology is that existing systems are big and unwieldy: their basic functionality is often concealed behind a thick layer of GUI, and their algorithms are not documented (they are trade secrets). This makes it difficult to use commercial systems in a research setting, or even to reverse-engineer their functionality.
- The obvious alternative is to re-implement the TM functionality from scratch.

Translation Memory
Conceptually, a Translation Memory consists of:
- a database containing all source-language segments (typically a sentence) and its translation in the target language;
- a similarity function f; and
- a filtering threshold α.

Given a new sentence to translate q (the query), the core TM functionality consists in finding the best match for q in D, i.e., the pair (s, t) with maximum similarity s \cdot f (q, s) / \alpha, then the system outputs the target-language counterpart of s, otherwise nothing.

Similarity Function f measures the similarity between two source-language strings. Typically, it produces a value between 0 and 1, where 0 means "completely different" and 1 means "identical"; α can then be in the range [0, 1]. It is generally acknowledged that commercial TM systems use variants of the Levenshtein distance, e.g.,

\text{Levenshtein}(q, s) = \min[α \cdot \text{cost}(q, s)]

MT Evaluation Metrics

Implementing the core TM functionality requires that we come up with a similarity function. As it turns out, the sub-field of TM research has churned out many such functions: MT evaluation metrics. Many (if not all) of the MT evaluation metrics proposed in recent years rely on measuring the similarity between a machine translation output and one or more reference translations. In this study, we examine five different evaluation metrics:

- **WER** - Word-error rate is based on word-level Levenshtein distance. As far as anyone knows, this is essentially what is used in commercial TM systems, and serves as baseline for this study.
- **BLEU** - Papineni et al. (2002) - based on n-gram precision, it implements the idea of accounting separately for adequacy (low-order n-grams) and fluency (high-order n-grams).
- **METEOR** - Denkowski & Lavie (2011) - linguistic resources (stemmer, WordNet, paraphrases) bring us closer to semantic similarity.
- **CMETEOR** - A Poor Man's Translation Memory Using Machine Translation Evaluation Metrics

We implement an exhaustive search strategy: for every pair (s, t) in the TM D, measure the similarity between q and s, using MT evaluation metric X’s similarity function:

\text{similarity(q, s)} = \max [\alpha \cdot f(q, s)]

Results

This table reports which similarity function (q, s) produces best, according to each MT evaluation metric. Who wins the race depends heavily on who is keeping the score!

**Evaluation Methodology**
We opt for the evaluation approach proposed in Simard & Isabelle (2009), in which TM systems are evaluated as if they were MT systems: test sentences are submitted to the TM, with the filtering threshold \alpha set to zero, thus effectively inhibiting output filtering. The target segments of the best matches are then compared to the reference translations, using standard MT evaluation metrics. In practice, in this study, we use the same metrics that were used as similarity functions.

Data
We perform experiments to assess the performance of each MT evaluation metric as TM similarity function. Experiments were done using English, French, German, and Spanish data, drawn from Europarl v6 (Koehn, 2005), the OPUS corpus (Tiedemann, 2009) and the JRC-Acus (v2.2, Steiner et al., 2004). From each corpus, we randomly sampled 1000 pairs of segments, to be used as test data; the rest was used to build translation memories.

<table>
<thead>
<tr>
<th>Corpus</th>
<th>Language</th>
<th>TM (“raw”) Test words</th>
<th>Evaluation metric</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>WER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BLEU</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NIST</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>METEOR</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CMETEOR</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Source Ratio</th>
<th>Target Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.85</td>
<td>1.48</td>
</tr>
<tr>
<td>±0.58</td>
<td>±2.07</td>
</tr>
<tr>
<td>1.01</td>
<td>1.04</td>
</tr>
<tr>
<td>1.03</td>
<td>1.06</td>
</tr>
<tr>
<td>1.31</td>
<td>1.43</td>
</tr>
<tr>
<td>1.56</td>
<td>1.45</td>
</tr>
</tbody>
</table>


**Results**

- **When WER is used to measure precision, WER always comes out as the best similarity function.**
- **BLEU generally exhibits similar behavior.**
- **VMETEOR and METEOR always prefer one of the MT family; both metrics always agree with one another, usually preferring WER when the source language and METEOR when English is target.**
- **NIST has low self-esteem: it is because local optimization (finding the best match for each segment) doesn’t guarantee a global maximum.**

**Conclusions**
MT evaluation metrics can be used effectively as translation memory similarity functions. Each metric has its own characteristics and potential benefits, but evaluation is problematic. Metrics based on n-gram precision such as BLEU and NIST are less computationally expensive than classic edit-distance-based metrics such as WER, or metrics that rely on linguistic resources, such as METEOR. In practice, they are easy to implement and produce results comparable to WER, especially in high-similarity, where it counts for real TM usage.

- **Customizing linguistic resources such as paraphrase tables could help in better long-term quality.** Human-based evaluation may be only credible alternative, and is what we plan to resort to in future experiments.