A Flexible Online Server for Machine Translation Evaluation

Matthias Eck, Stephan Vogel, and Alex Waibel
InterACT Research
Carnegie Mellon University
Pittsburgh, PA, 15213, USA
{matteck, vogel, waibel}@cs.cmu.edu

Abstract. We present an Online Server for Machine Translation Evaluation that offers improvements over the standard usage of the typical scoring scripts. Users are able to interactively define their own test sets, experiments and pre-processing steps. Several scores are automatically calculated for submitted translations and the hypotheses and scores are organized and archived for later review. The server offers a nice web based user interface.

1. Introduction

Evaluating machine translation hypotheses is a very important part of the ongoing research. Automatic scoring metrics allow a fast evaluation of translations and a quick turn-around for experiments. Researchers rely on the evaluation metrics to measure performance improvements gained by new approaches.

The well-known automatic scores for machine translation are BLEU and NIST but a variety of other scores is available (Papineni, Roukos, Ward, and Zhu, 2002; Doddington, 2001; Banerjee and Lavie, 2005). Most of the scores rely on software programs or scripts that expect a variety of parameters including the hypothesis and reference files. The software then calculates the appropriate score. For most applications the files have to be in a special SGML file format that tags the different parts of the hypothesis or reference file.

It is especially difficult for newcomers or for people who just want to get a glimpse of the possibilities to use these software programs. An experienced developer will most probably have a sophisticated setup for translation scoring but this will take a while for a beginner.

The web server application presented here tries to circumvent some of the difficulties of scoring machine translation output. The online user interface offers an interactive environment in which test sets and experiments can be defined and hypotheses can be scored. The server stores the submitted translations for later review. It also offers directly accessible web services that allow score calculation in scripts and software programs based on the defined test sets.

2. Related Work

Online Servers for Competitive Evaluations

Different online servers have been used to evaluate translations for a variety of competitive evaluations. Especially notable are the evaluation servers for the NIST MT Evaluations (NIST, 2001-2006) and for the Evaluations in the International Workshops for Spoken Language Translation (IWSLT) in the years 2004 and 2005[1] (Akiba, Federico, Kando, Nakaiwa, Paul, and Tsuji, 2004; Eck and Hori, 2005). All of these evaluation servers were geared towards the needs of a competitive evaluation. The main goal was to make it easier for the organizers to handle a large amount of translation submissions and not necessarily to support the research of the participants. The servers did for example not show any scores during the actual evaluation period so that tuning the systems was impossible. The servers also did not provide any possibility to the participants to set up their own test sets.

[1] The server presented here was developed based on the server used for IWSLT 2005.

Evaluation Application

Another similar work is the EvalTrans tool presented in Nießen, Och, Leusch, and Ney (2000). Here the focus is on a locally installed tool that allows better and faster human evaluation by having a nice interface to support the evaluators. This tool is able to automatically extrapolate known human scores to similar sentences and give a prediction of the actual human score. Automatic evaluation scores can also be calculated.

3. Standard Scoring Routine

SGML File Format

For most scoring software the first step is to convert the hypothesis (candidate translation), reference and sometimes source files into an SGML defined format. SGML here offers additional flexibility compared to standard text files, mainly, the possibility of having different reference translations for a given sentence. Figure 1 shows how a simple SGML-tagged hypothesis could look like (with the appropriate values filled in).

```
<TSTSET setid="setid" trglang="language" srclang="language">
<DOC docid="docid" sysid="sysid">
<SEG id=1>hypothesis sentence 1</SEG>
<SEG id=2>hypothesis sentence 2</SEG>
...
</DOC>
</TSTSET>
```

Figure 1: SGML tagged translation hypothesis

The main difference for an SGML-tagged reference file is <REFSET> that replaces <TSTSET> as the main tag. It is also possible to have different <DOC> tags within one file that can be used to provide more than one reference translation per sentence (see Figure 2). Some scripts also expect the original source file to be in SGML format.

```
<REFSET setid="setid" trglang="language" srclang="language">
<DOC docid="docid" sysid="reference1">
<SEG id=1>reference 1 sentence 1</SEG>
<SEG id=2>reference 1 sentence 2</SEG>
...
</DOC>
<DOC docid="docid" sysid="reference1">
<SEG id=1>reference 2 sentence 1</SEG>
<SEG id=2>reference 2 sentence 2</SEG>
...
</DOC>
</REFSET>
```

Figure 2: SGML tagged reference translation

Invoke Scoring Software

After this step the actual command to execute the scoring script is similar to:

```
$ scoretranslation -r referencefile -s sourcefile -t hypothesisfile
```

Most machine translation scoring procedures follow this setup with slight changes and possibly additional parameters and options.

Annoyances

While none of these steps is very inconvenient there are a number of little annoyances in the whole process as SGML files have to be prepared and scoring scripts downloaded and installed. It is also necessary to find out the correct usage of the scoring scripts via user unfriendly command line interfaces. Files tend to be distributed over several directories with long pathnames which makes it especially hard to find the translations after a couple of months. It is also necessary to make sure that the same preprocessing steps are always applied.

4. Server for Online Evaluation

4.1. Requirements

Typical researchers in machine translation will have a number of training and test sets. After implementing new ideas or changing any part of the pre- or post-processing, training or decoding they will compare the automatic scores on a test set with the baseline score. Systematically trying different parameter settings for the new approach and comparing the results leads to maximizing its impact.

While every experiment could use a different test set it is common practice to reuse test sets to be able to compare the scores to earlier experiments. The public availability of test sets from the well-known competitive evaluations also allows other researchers to easily compare published scores with their own results.

The goal of the server presented here is to support researchers during their work with fast automatic evaluation scores. The user should be able to define test sets, experiments and score translations without any need to know anything about the inner workings of the scoring scripts, their parameters or file formats. The application in its current form is mainly geared towards the support of machine translation research but could also be extended or used for other text based scores most notably evaluations of Automatic Speech Recognition systems.

4.2. Design and Implementation

The initial requirement is that there is a concept of a test set with reference and source that can be used to score translations. We also decided to add an additional layer of abstraction with the introduction of the “Experiment” concept. An “Experiment” consists of a test set and additional information about which preprocessing steps to take and which scores to calculate. But it also and especially serves as a means of organizing translations for different approaches that are using the same test set. The overall design is shown in Figure 1. This diagram illustrates the relationships and variables for the concepts “Test Set”, “Experiment” and “Translation”. The underlying database is modeled according to this diagram.

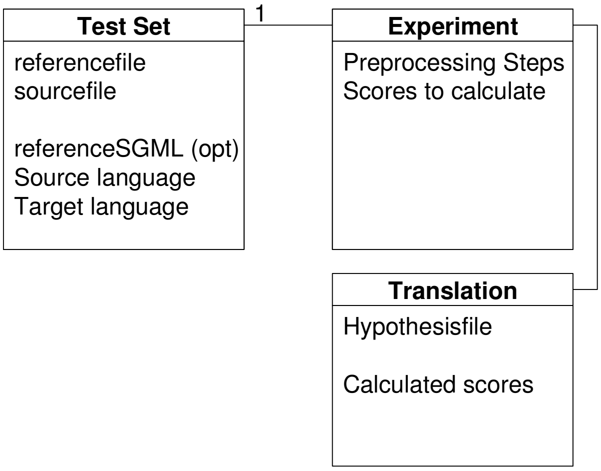


Figure 3: General design of the underlying data structure

Figure 4 shows the practical application of this design. The same test set can be used in three different experiments. Two of these experiments use the same preprocessing while the third experiment applies different preprocessing.

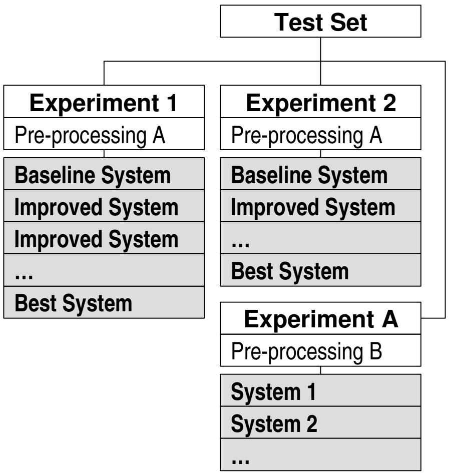


Figure 4: Example Test Set used in 3 Experiments

User Interface and Web Services

The online user interface is intended to be clean and simple and to give the user easy and intuitive access to the functions of the server. The use of the web interface will be described in Section 4.3.

A more advanced way to access the functions has also been implemented. While a web interface is very convenient for the user, it is very hard to use from scripts or programs involved to produce a number of translations. Thus, there is also a direct way to score translations using typical programming languages with predefined test sets. The web service technology offers an easy way to accomplish this. Using the SOAP protocol a web server can provide functions that every programming language with the necessary SOAP libraries can directly access. An example on invoking web services will be given at the end of section 4.3.

Implementation Considerations

The server has been implemented as a web application using PHP scripts on an Apache webserver. The database used is MySQL. The scoring scripts mainly use Perl and are directly called from within the PHP scripts.

4.3. Practical application

General Information

The scoring server is available at:

http://www.is.cs.cmu.edu/mteval

All major web browsers can be used to access this website. The following description is limited to the most important functions. For a more in depth description please check the web site for the latest documentation.

Supported Scores

The scoring server right now supports the calculation of the following scores:

- BLEU (Papineni et al., 2002)
- NIST (Doddington, 2001) version 11b, available from http://www.nist.gov/speech/tests/mt/resources/scoring.htm
- 95% confidence intervals for NIST and BLEU scores based on 1000 samples (Zhang and Vogel, 2004).
- mWER, mPER (word and position independent error rate based on multiple references).
- METEOR (Banerjee and Lavie, 2005) version 0.4.3, available from http://www.cs.cmu.edu/~alavie/METEOR/

The user can select any combination of these scores. Especially the confidence intervals can take some time to compute so it might be reasonable to not calculate those for every translation submitted. Missing scores can simply be recalculated for interesting submissions (e.g. baseline, best systems).

Additional scoring metrics can be added to the application if they support the standard SGML format with multiple references. Feedback from users will be especially appreciated here.

Registering a New User

First a new user has to be registered. After entering the required information and a username and password the user gets access to the evaluation server.

Main Functions

The main menu on top of the screen offers the three main functions offered by the server:

- Submit Translation
- Define Experiments
- Define Test sets

It also offers the administrative functions to edit the user information and to log out.

Defining New Test Sets

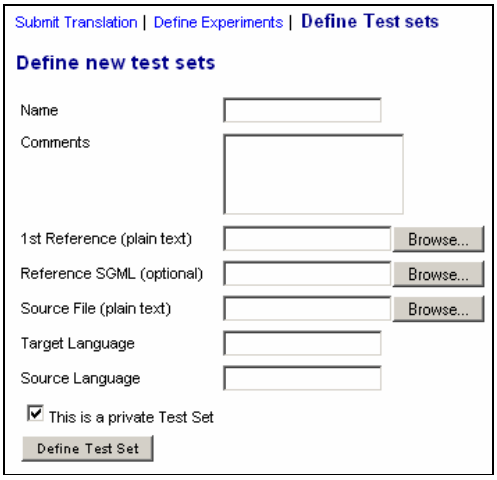
A new user will not have any private test sets yet, so the first step is to define a new test set. A new user will however have access to test sets that were defined as public by other users. The form to define a new test set is shown in Figure 5. A test set is identified by its name. The user also has the option to give additional comments. The first reference translation and the source file have to be uploaded in plain text and the target and source language have to be identified. If it is intended to use more than one reference, an additional reference SGML file can be uploaded as well.

The test set can either be private or public. A private test set can only be accessed and used by the user who originally defined it while a public test set is accessible by every user. It is necessary to ensure that there are no copyright limitations before a test set can be public.

Defining New Experiments

After a test set has been defined it can be used to define a new experiment (Form in Figure 6).

An experiment is also identified by a name and the users select one of their test sets as the basis for this experiment.



Submit Translation | Define Experiments | Define Test sets

Define new test sets

Name

Comments

1st Reference (plain text) Browse...

Reference SGML (optional) Browse...

Source File (plain text) Browse...

Target Language

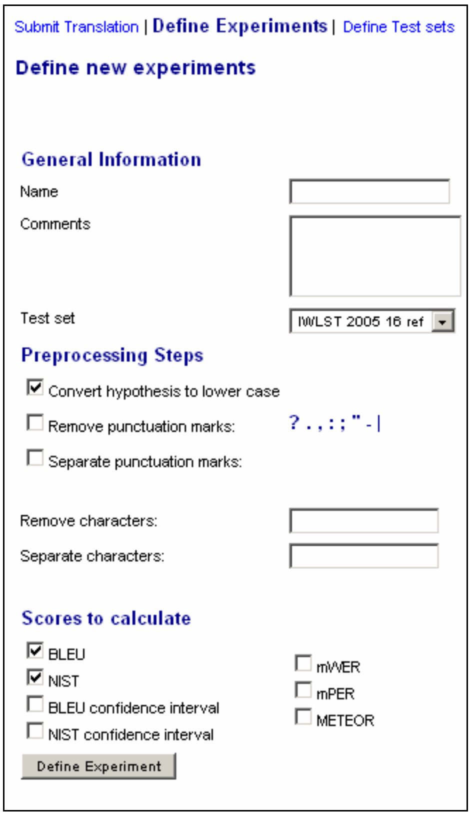
Source Language

☑ This is a private Test Set

Define Test Set

Figure 5: Form to define new test sets

The next step is to define the pre-processing of the uploaded candidate translations. It is possible to convert the hypothesis to lower case and remove or separate the standard punctuation marks. The users can also enter arbitrary characters that should be removed or separated. This will be especially useful for languages with a different set of punctuation marks. In the last part of the form the user selects which scores should be calculated for this experiment.



Submit Translation | Define Experiments | Define Test sets

Define new experiments

General Information

Name

Comments

Test set: IWLST 2005 16 ref

Preprocessing Steps

☑ Convert hypothesis to lower case

☐ Remove punctuation marks: ? . , : ; " - |

☐ Separate punctuation marks:

Remove characters:

Separate characters:

Scores to calculate

☑ BLEU ☐ mWER

☑ NIST ☐ mPER

☐ BLEU confidence interval ☐ METEOR

☐ NIST confidence interval

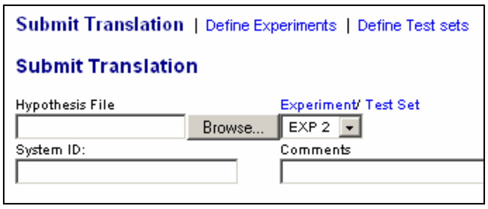
Define Experiment

Figure 6: Form to define new experiments

Submitting Translation Hypotheses

Finally with a defined experiment it is possible to submit actual translation hypotheses and calculate the selected scores (Figure 7).

After the translation hypothesis has been submitted the server will calculate the requested scores for the selected experiment. After all scores have been calculated the new hypothesis will show up in the list of submitted hypotheses with the respective scores.



Submit Translation | Define Experiments | Define Test sets

Submit Translation

Hypothesis File Browse...

Experiment/ Test Set: EXP 2

System ID:

Comments

Figure 7: Form to submit translation hypotheses

Archiving of Previous Scores

This view gives the user a summary of the submitted translations and scores. It is also possible to calculate other scores by clicking on the "-calc-" links or to directly compare the hypothesis with the first reference by clicking on the hypothesis filename. This automatic archiving of the previously calculated scores and the respective hypotheses is one of the main advantages of the server presented here. Figure 9 shows an example overview with 3 different experiments and a number of submissions for each experiment.

Usage via Web Services

The web services defined for this online evaluation server allow a direct call of the scoring functions from virtually any programming language.

The following example (Figure 8) in PHP uses the NuSOAP module to call the provided function. The web site interface will provide the necessary testsetid. For more detailed descriptions please consult the online documentation.

```
//Load hypothesis file from disk
$file="hypothesis";
$tstFileHandle = fopen($file,"r");
$tstFileContent = fread($tstFileHandle, filesize($file));
```

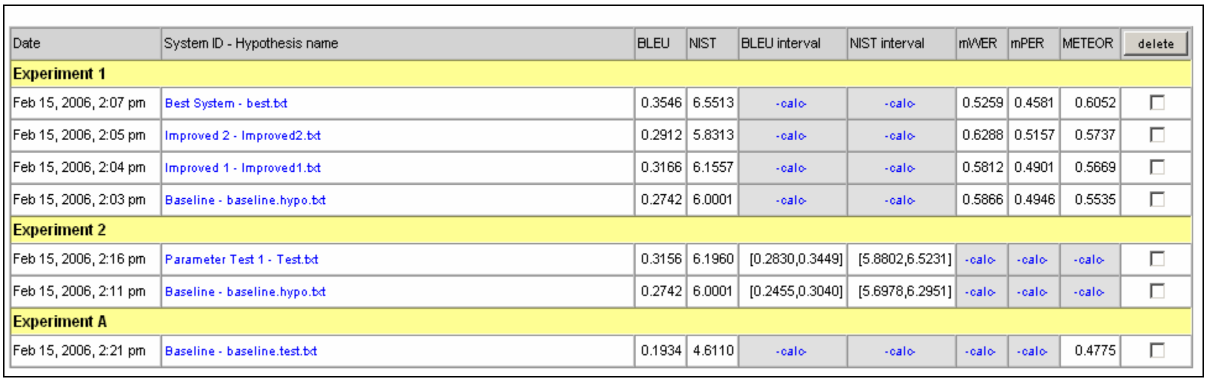
```
//Define necessary parameters
$parameters = array(          'hypothesis' => $tstFileContent,
                         'testsetid' => testsetid
                         'score'=>'BLEU', );
//Connect to Web Service
$soapclient = new soapclient_nusoap
('http://moon.is.cs.cmu.edu:8080/EvaluationServer2/
webservice.php');
//Call Web Service
$score=$soapclient->call('score',$parameters);
```

Figure 8: Web service invocation with PHP

5. Conclusions

The web application presented here offers a convenient interface for the evaluation of machine translation hypotheses compared to the standard techniques. The functions are also available via web services for handy usage in typical programming languages.

We intend to continue to further improve the server by adding other scores and giving more detailed outputs as well as improved statistical analysis. The hope is that with more feedback we will get a better understanding of what the users actually expect from such a tool and we will try to incorporate those findings.



Date	System ID - Hypothesis name	BLEU	NIST	BLEU interval	NIST interval	mWER	mPER	METEOR	delete
Experiment 1									
Feb 15, 2006, 2:07 pm	Best System - best.txt	0.3546	6.5513	-calc-	-calc-	0.5259	0.4581	0.6052	□
Feb 15, 2006, 2:05 pm	Improved 2 - Improved2.txt	0.2912	5.8313	-calc-	-calc-	0.6288	0.5157	0.5737	□
Feb 15, 2006, 2:04 pm	Improved 1 - Improved1.txt	0.3166	6.1557	-calc-	-calc-	0.5812	0.4901	0.5669	□
Feb 15, 2006, 2:03 pm	Baseline - baseline.hypo.txt	0.2742	6.0001	-calc-	-calc-	0.5866	0.4946	0.5535	□
Experiment 2									
Feb 15, 2006, 2:16 pm	Parameter Test 1 - Test.txt	0.3156	6.1960	[0.2830,0.3449]	[5.8802,6.5231]	-calc-	-calc-	-calc-	□
Feb 15, 2006, 2:11 pm	Baseline - baseline.hypo.txt	0.2742	6.0001	[0.2455,0.3040]	[5.6978,6.2951]	-calc-	-calc-	-calc-	□
Experiment A									
Feb 15, 2006, 2:21 pm	Baseline - baseline.test.txt	0.1934	4.6110	-calc-	-calc-	-calc-	-calc-	0.4775	□

Figure 9: Translation Score Overview Table

6. References

AKIBA, Yasuhiro, FEDERICO, Marcello, KANDO, Noriko, NAKAIWA, Hiromi, PAUL, Michael and TSUJI, Jun'ichi (2004). 'Overview of the IWSLT04 Evaluation Campaign'. Proceedings of IWSLT 2004, Kyoto, Japan.

BANERJEE, Satanjeev, LAVIE, Alon (2005). 'METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments'. ACL 2005 Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, Ann Arbor, MI, USA.

DODDINGTON, George (2001). 'Automatic Evaluation of Machine Translation Quality using n-Gram Co-occurrence Statistics'. NIST Washington, DC, USA.

ECK, Matthias and HORI, Chiori (2005). 'Overview of the IWSLT 2005 Evaluation Campaign'. Proceedings of IWSLT 2005, Pittsburgh, PA, USA.

NIEßEN, Sonja, OCH, Franz Josef, LEUSCH, Gregor, and NEY, Hermann (2000). 'An Evaluation Tool for Machine Translation: Fast Evaluation for MT Research'. In Proceedings of 2nd International Conference on Language Resources and Evaluation (LREC 2000), Athens, Greece.

NIST MT Evaluation information and schedule http://www.nist.gov/speech/tests/mt/schedule.htm National Institute of Standards and Technology, 2001-2006.

NuSOAP SOAP Toolkit for PHP http://sourceforge.net/projects/nusoap/

PAPINENI, Kishore, ROUKOS, Salim, WARD, Todd, and ZHU, Wei-Jing (2002). 'BLEU: a Method for Automatic Evaluation of Machine Translation'. Proceedings of ACL 2002, Philadelphia, PA, USA.

ZHANG, Ying and VOGEL, Stephan (2004). 'Measuring Confidence Intervals for the Machine Translation Evaluation Metrics'. Proceedings of the International Conference on Theoretical and Methodological Issues in Machine Translation (TMI 2004), Baltimore, MD, USA.