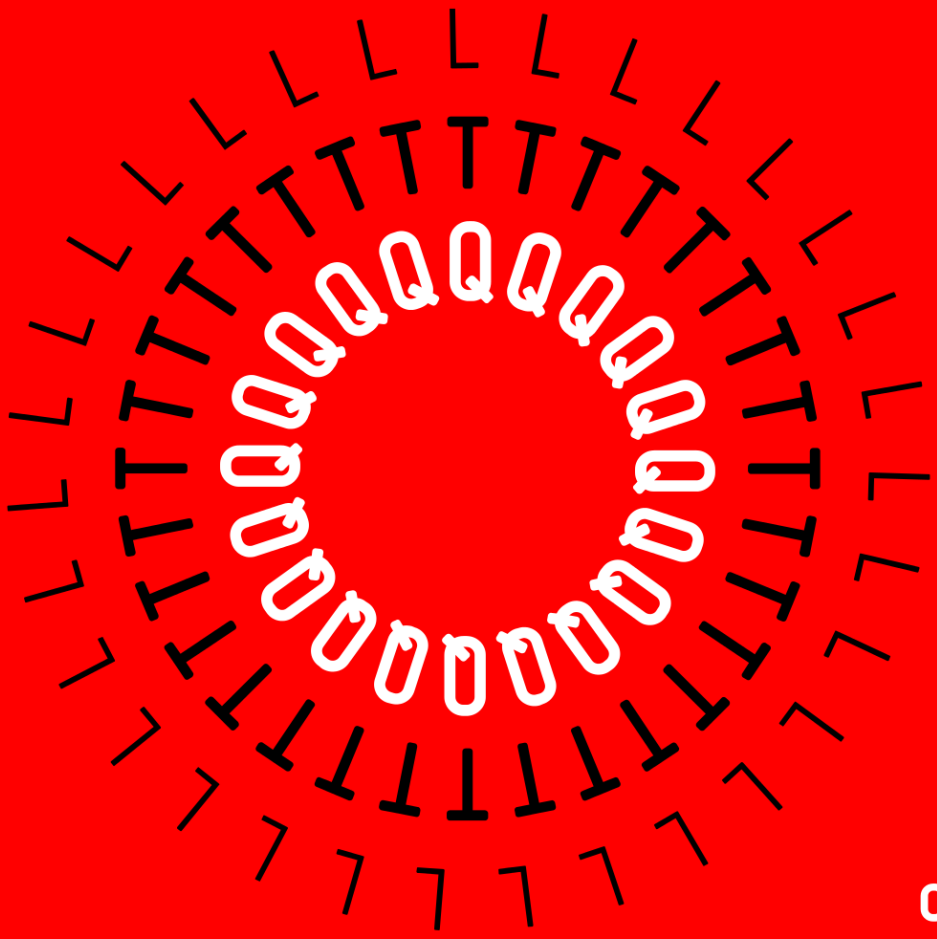


qtleap
quality translation by deep language engineering approaches

Report on MT improved with semantic linking and resolving

DELIVERABLE D5.11

VERSION 1.3 | 2016-11-9

QTLeap

Machine translation is a computational procedure that seeks to provide the translation of utterances from one language into another language.

Research and development around this grand challenge is bringing this technology to a level of maturity that already supports useful practical solutions. It permits to get at least the gist of the utterances being translated, and even to get pretty good results for some language pairs in some focused discourse domains, helping to reduce costs and to improve productivity in international businesses.

There is nevertheless still a way to go for this technology to attain a level of maturity that permits the delivery of quality translation across the board.

The goal of the QTLeap project is to research on and deliver an articulated methodology for machine translation that explores deep language engineering approaches in view of breaking the way to translations of higher quality.

The deeper the processing of utterances the less language-specific differences remain between the representation of the meaning of a given utterance and the meaning representation of its translation. Further chances of success can thus be explored by machine translation systems that are based on deeper semantic engineering approaches.

Deep language processing has its stepping-stone in linguistically principled methods and generalizations. It has been evolving towards supporting realistic applications, namely by embedding more data based solutions, and by exploring new types of datasets recently developed, such as parallel DeepBanks.

This progress is further supported by recent advances in terms of lexical processing. These advances have been made possible by enhanced techniques for referential and conceptual ambiguity resolution, and supported also by new types of datasets recently developed as linked open data.

The project QTLeap explores novel ways for attaining machine translation of higher quality that are opened by a new generation of increasingly sophisticated semantic datasets and by recent advances in deep language processing.

www.qtleap.eu

Funded by
QTLeap is funded by the 7th Framework Programme of the European Commission.

Supported by
And supported by the participating institutions:

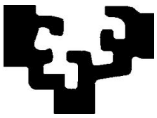
Faculty of Sciences, University of Lisbon

German Research Centre for Artificial Intelligence

Charles University in Prague

Bulgarian Academy of Sciences

Humboldt University of Berlin



University of Basque Country

University of Groningen

Higher Functions, Lda

Revision history

Version	Date	Authors	Organisation	Description
0.1	May 15, 2016	Eneko Agirre, Arantxa Otegi	UPV/EHU	First draft of improved tools section
0.2	Jun 11, 2016	Eneko Agirre	UPV/EHU	First draft of the MT experiments section
0.3	Jul 15, 2016	Arantxa Otegi, Eneko Agirre	UPV/EHU	Redo structure
0.4	Aug 5, 2016	Arantxa Otegi	UPV/EHU	Section 2
0.5	Sep 9, 2016	Eneko Agirre	UPV/EHU	Exec. Summary and Introduction
0.6	Sep 15, 2016	Gorka Labaka	UPV/EHU	Section 2
0.7	Sep 17, 2016	João Silva	FCUL	Section 2
0.8	Sep 19, 2016	Chakaveh Saedi, João Rodrigues	FCUL	Sections 4, 5
0.9	Oct 10, 2016	Nora Aranberri, Eneko Agirre	UPV/EHU	Section 4
1.0	Oct 29, 2016	Martin Popel, Gorka Labaka	CUNI, UPV/EHU	Section 3
1.1	Oct 29, 2016	Kiril Simov, Petya Osenova	IICT-BAS	Section 3
1.2	Oct 30, 2016	Eneko Agirre, Gorka Labaka	UPV/EHU	Overall review
1.3	Oct 30, 2016	Eneko Agirre, Gorka Labaka, Kiril Simov	UPV/EHU, IICT-BAS	Changes after internal review

Statement of originality
This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

Report on MT improved with semantic linking and resolving

DOCUMENT QTLEAP-2016-D5.11
EC FP7 PROJECT #610516

DELIVERABLE D5.11

completition
FINAL
status
SUBMITTED
dissemination level
PUBLIC

responsible
Eneko Agirre (WP5 Coordinator)
reviewer
Gertjan van Noord
contributing partners
UPV/EHU, FCUL, CUNI, IICT-BAS

authors
Eneko Agirre, Arantxa Otegi, Gorka Labaka, Chakaveh Saedi, João Rodrigues, João Silva, Martin Popel, Roman Sudarikov, Kiril Simov, Petya Osenova

Contents

Executive summary

The goal of the QTLeap project is to develop Machine Translation (MT) technology that goes beyond the state of the art in terms of "depth" of the methods and knowledge used. The goal of WP5 is to enhance MT with advanced crosslingual methods for the resolution of referential and lexical ambiguity by pursuing the following objectives:

1. to provide for the assembling and curation of the language resources and tools (LRTs) available to support the resolution of referential and lexical ambiguity (Task 5.1, starting M1);

2. to leverage the resolution of referential and lexical ambiguity by means of advanced crosslingual named-entity-resolution and word-sense-resolution methods (Task 5.2, starting M1);

3. to proceed with the intrinsic evaluation of the solutions found in the previous task (Task 5.3, starting M10);

4. to contribute for high quality machine translation by using semantic linking and resolving to improve MT (Task 5.4, starting M17). In particular Pilot 2 (M24) will be devoted to check the contribution of the tools in this WP to MT.

The work reported in this document has been carried out along the plans and is based on the project Description of Work (DoW) and Deliverable 5.1 ("State of the art").

Deliverable D5.11 aims at providing a final report on the improvements in Machine Translation (MT) related to the semantic linking and resolving activities in WP5 (work package 5). The main activities refer to the resources and tools integrated in the latest QTLeap machine translation engine (Pilot 3, described in Deliverable D2.11), which following the planning in the DoW, includes the following experiments in Task 5.4:

- Online gathering of multilingual information, Experiment 5.4.4.

- New transduction algorithms, Experiment 5.4.5.

Deliverable D5.9 reported the final version of the LRTs (Tasks 5.1), as released in D5.8, as well as the evaluation of the advanced processors (Tasks 5.2 and 5.3). This deliverable reports the latest improvements of English NED since then.

1 Introduction

Deliverable D5.11 aims at providing a final report on the improvements in Machine Translation (MT) related to the semantic linking and resolving activities in WP5 (work package 5). These activities include linguistic processors like Word Sense Disambiguation (WSD), Named-Entity Disambiguation (NED) and Coreference resolution, and Linked Open Data (LOD) resources like WordNet or DBpedia[1] (the LOD version of Wikipedia) for the languages covered in WP5: Basque, Bulgarian, Czech, English, Portuguese and Spanish.[2] The strategy was to explore and experiment with techniques that help to improve Machine Translation performance, and carry over the successful ones to the final QTLeap system, Pilot 3.

The main activities refer to the resources and tools integrated in the latest QTLeap machine translation engine (Pilot 3, described in Deliverable D2.11), which include:

- Online gathering of multilingual information, Experiment 5.4.4, in Section 2.
- New transduction algorithms, Experiment 5.4.5, in Section 3.

This report also includes additional contents (Section 4), including the extension of the positive results of WSD for Portuguese to other languages, the application of WSD to Bulgarian, the analysis of the potential contribution of coreference when translating to Basque and Spanish, and the improvement of the NED advanced processor.

The deliverable is structured as follows. Section 2 reports experiment 5.4.4 and Section 3 reports experiment 5.4.5. Section 4 describes the additional experiments on WSD, coreference and NED. Section 5 summarizes the success of experiments, and reports on their integration in Pilot 3. Finally, Section 6 draws the final remarks.

[1] http://dbpedia.org

[2] Although not planned in the DoW, some experiments reported in this deliverable also cover Dutch.

2 Experiments 5.4.4: Online gathering of multilingual information

The goal of Experiment 5.4.4 is to handle unknown expressions by resorting to information sources of multilingual information whose content evolves very rapidly and is being constantly growing. We first explored online comparable corpora that we could use for the experimentation, analyzing its availability and suitability for the purpose of the experiment:

Wikipedia We find it to be an adequate multilingual comparable corpora, as it has many articles related to information technology (IT).

DBpedia As it is the Linked Open Data version of Wikipedia, it could be a good alternative to Wikipedia. On the plus side, it has an online server, on the negative side it does not provide any additional terminology, and it only serves article titles, missing the full text of the articles.

Twitter Our preliminary analysis showed that gathering tweets related to the IT domain would not be straightforward, and would require to mine all tweets. Moreover, collecting data in other languages than English would be difficult, because of the lower traffic.

Blogs We considered using one of the many IT related blogs available online, such as IT-related posts of the Stack Overflow Q&A site[3]. Although this might be a good source for some subdomains of IT for English, the coverage of other languages[4] is very sparse, and it would be difficult to find the amount of data needed.

Taking into account the above analysis, we went for Wikipedia, and thus, we designed an experiment to analyze whether comparable corpora from Wikipedia can be used to produce domain-relevant gazetteers (cf. Section 2.1 below).

In addition, we explored how to overcome some limitations on the TectoMT method to handle terms in Gazetteers. Rather than treating bilingual gazetteer entries as fixed forms, we explicitly represent their internal syntactic structure via treelets and integrate them in the t-layer of TectoMT (cf. Section 2.2 below).

2.1 Adding Wikipedia terminology as a gazetteer

Deliverable D5.7 already reported a closely related approach which was used in Pilot 2 (Experiment 5.4.3, described in Section 4.2 in D5.7). In that work the category structure of Wikipedia was used to identify IT-related articles. The titles of Wikipedia articles are the English terms and the inter-Wikipedia links are used to find their corresponding translations to other languages. In this experiment we follow a similar strategy to identify IT-related Wikipedia articles, but instead of limiting ourselves to the titles of the articles, we automatically extract bilingual terminology from the text of the selected Wikipedia articles.

Note also that we investigated a more sophisticated method to identify IT-related articles. Experiment 5.4.3 adopted the method by Gaudio and Branco [2012] to mine IT-related articles, using as starting point the most generic categories in the IT field, and then,

[3] http://stackoverflow.com/

[4] See for instance http://es.stackoverflow.com/

English title	Spanish title
External sorting	Ordenamiento externo
Quickoffice	Quickoffice
ASCII	ASCII
PowerBuilder	PowerBuilder
AIMP	AIMP
Modular programming	Programación modular
OpenSearch	OpenSearch
Alureon	Alureon
Binary blob	Blob binario
Certified Information Systems Auditor	CISA
Priority inversion	Inversión de prioridades
Advanced Audio Coding	Advanced Audio Coding
Shell (computing)	Shell (informática)
Fraps	FRAPS
Amiga	Commodore Amiga
WorkNC	WorkNC
EA Black Box	EA Black Box
Software crisis	Crisis del software
NComputing	NComputing
RanXerox	RanXerox

Table 1: Parallel titles of Wikipedia articles extracted with WikiTailor (EN-ES).

all the articles linked to these categories and their children were selected. The threshold was simply the number of articles we wanted to extract. The experiment presented here, instead, follows the work by Barrón-Cedeño et al. [2015], where a more complex method to select IT-related articles is set forth, which tries to avoid articles that are not related to the IT domain.

The term extraction consisted of two steps:

1. Extract parallel titles of articles relevant to the IT domain using WikiTailor [Barrón-Cedeño et al., 2015], in two variants (union and intersection). Once we have the parallel titles, get the comparable corpora, that is, texts from the corresponding Wikipedia articles.

2. Extract bilingual terminology from the text of the comparable corpora obtained in the previous step, using TermSuite[5]. TermSuite is a toolbox for terminology extraction and multilingual term alignment developed by TTC European project[6].

A manual inspection of the article titles obtained with WikiTailor and the domain terms for the gazetteers obtained with TermSuite shows that they are of good quality. Table 1 shows a sample the article titles mined by Wikitailor for English and Spanish. Table 2 shows bilingual terminology extracted with TermSuite (top 40 terms from the list of multiword terms).

In order to measure the quality of the automatically extracted bilingual terminology we checked the effect of such a terminology when it is used as a gazetteer in the EN-

[5] http://termsuite.github.io/
[6] http://www.ttc-project.eu/

English term	Spanish term
search algorithm	algoritmo de búsqueda
quantum algorithm	algoritmo cuántico
numerical analysis	análisis numérico
hash function	función hash
genetic algorithms	algoritmo genético
euclidean algorithm	algoritmo de euclides
computer vision	visión artificial
bubble sort	ordenamiento de burbuja
image processing	retoque fotográfico
game console	videoconsola
formal language	lenguaje formal
desktop publishing	autoedición
decision problem	problema de decisión
adobe photoshop	adobe photoshop
wikimedia foundation	fundación wikimedia
video game	videojuego
video game console	videoconsola
video game consoles	videoconsola
user agent	agente de usuario
stochastic neural networks	red neuronal estocástica
source code editor	editor de código fuente
single user mode	monousuario
recommender systems	sistema de recomendación
programming language	lenguaje de programación
primitive data types	tipo de dato elemental
man-in-the-middle attack	ataque man-in-the-middle
knowledge base	base de conocimiento
hypertext transfer protocol	hypertext transfer protocol
home page	página de inicio
hidden markov models	modelo oculto de márkov
floating point	coma flotante
filename extension	extensión de archivo
computer monitor	monitor de computadora
binary files	archivo binario
audio file formats	formato de archivo de audio
algebraic data types	tipo de dato algebraico
web traffic	tráfico web
web services	servicio web
web pages	página web
web development	desarrollo web

Table 2: Bilingual terminology extracted with TermSuite (EN-ES).

	no other gazetter	localization	MTC+localization
no Wikipedia gazetteer	29.60	31.61	31.98
Pilot2 Wikipedia gazetteer	29.84	31.61	32.25
WikiTailor union	29.83	31.84	–
WikiTailor interset	**29.95**	31.78	–
Termsuite	29.60	32.01	–
all Wikipedia gazetteers	29.86	**32.05**	**32.26**

Table 3: The BLEU scores (Batch2a) of the en→es translation using several gazetteers. In bold, the best results in each column.

	no other gazetter	localization	MTC+localization
no Wikipedia gazetteer	25.98	24.41	23.83
Pilot2 Wikipedia gazetteer	26.50	25.62	24.82
WikiTailor union	26.48	**25.86**	–
WikiTailor interset	26.21	24.95	–
Termsuite	25.98	24.41	–
all wikipedia gazetteers	**26.66**	25.81	**24.84**

Table 4: The BLEU scores (Batch2q) of the es→en translation with and without using gazetteers. In bold, the best results in each column.

ES TectoMT system[7]. We evaluated four gazetteers extracted from Wikipedia: the one used in Pilot2 (cf. Section 4.2 in D5.7), the list of parallel titles obtained with *union* and *interset* configurations of WikiTailor, and the one extracted from the comparable article texts with Termsuite. Table 3 shows the results on the Batch2a dataset of the en→es translation measured by BLEU. In the rows, we report the results obtained when none of the Wikipedia gazetteers is used, with the effect of each of them individually and the combined effect of all of them. In order to check any possible interference with other gazetteers, the columns report the numbers when combined with the Localization gazetteers used in Pilot2 (incl. KDE, LibreOffice, VLC, cf. Section 4.2 in D5.7) and the gazetteer construct from the publicly available Microsoft Terminology Collection (MTC)[8].

The results in the first column show that the results among the four Wikipedia gazetteers are very similar, with only tiny gains for the more sophisticated WikiTailor and Termsuite Wikipedia-based gazetteers. The best results when combining the Wikipedia-based gazetteers with the localization gazetteers (second column) are for Wikitailor and for using all Wikipedia-gazetteers, but the gains are small. Finally, the best results are obtained when all gazetteers are combined (rightmost column), but we get almost a similar result when the only Wikipedia gazetteer is the Pilot 2 gazetteer (32.26 compared to 32.25). These results show that the terminology extracted with the more complex WikiTailor and Termsuite are mildly effective, but the results are very similar to the version already introduced in Pilot2.

Table 4 shows the results on the Batch2q dataset of the es→en translation. Similarly as in the previous dataset, TermSuite does not improve results over the baseline (25.98), although the combination of all Wikipedia gazetteers yields the best results (26.66). Contrary to Btach2a on the en→es direction localization and Microsoft gazetteers slightly degrade results. These results confirm that the more complex Wikitailor and Termsuite

[7]We used a version of TectoMT which is close to Pilot 2, although the treelet module was not activated.
[8]http://www.microsoft.com/Language/en-US/Terminology.aspx

are mildly effective, although they perform very close to the Wikipedia gazetteer already introduced in Pilot2.

The qualitative inspection of the bilingual terminology extracted from the Wikipedia-based comparable corpora using TermSuite shows good quality, although the integration with the rest of the gazetteers did only provide marginal improvements. In fact, the simpler method to extract IT-related terminology presented in D5.7 and tested in Pilot2 is similarly effective. Consequently, there is no need to include additional gazetteers in Pilot 3. Some of the reasons for the disappointing results could be related to the test suite of the MT experiments, based on PCMEDIC corpus content. The PCMEDIC corpus contains terms which are associated to end-user software. While the WikiTailor and TermSuite gazetteers are of good quality, they tend to include IT terms which are not associated to end-user software. We think that a different domain and user-case might show clearer improvements in translation quality when using WikiTailor and TermSuite.

2.2 Adding terminology as a transfer dictionary with treelets

As shown in the previous section, simple string matching with gazetteers is appropriate to translate fixed terms in the IT domain like menu items, button names and system messages. However, this technique has two important limitations when applied to terminology beyond fixed terms, e.g. terms including common nouns (e.g. driver, file) or verbs (e.g. run, set up):

1. It does not handle inflection, neither in the source language nor in the target language, so the different surface forms of a given term (e.g. run, runs, running, ran) will not be translated unless there is a separate entry for each of them. This is particularly relevant for morphologically rich languages like Spanish (specially in verb inflection), Basque and Czech.

2. It does not handle morphosyntactic ambiguity, for instance, the translation of the English term “test” can depend on it being used as a noun or as a verb.

In order to overcome these issues, we developed a terminology translation module with treelets, tree-like representations of single word and multiword terms, which are applied on the t-layer of TectoMT instead of the surface. The translation process involves the following steps:

1. **Preprocessing:** The terminology dictionary is first preprocessed so it can be efficiently used later at runtime. For that purpose, the lemma of each entry in the dictionary is independently analyzed up to the t-layer in both languages. This analysis is done without any context, so if there is some ambiguity, it might happen that the analysis given by the system does not match the sense it has in the dictionary. For instance, the English term ’file’ might be analyzed either as a verb or a noun, but its entry in the dictionary and, consequently, its translation, will correspond to only one of these senses. For that reason, we decide to remove all entries whose part-of-speech tag in the original dictionary does not match the one assigned to the root node by the analyzer.

2. **Matching:** During this stage, we search for occurrences of the dictionary entries in the text to translate, which is done at the t-layer. For that purpose, the preprocessed tree of a term is considered to match a subtree of the text to translate if the lemma

and part-of-speech tag of their root node are the same and their corresponding children nodes recursively match for all their attributes. By limiting the matching criteria of the root node to the lemma and part-of-speech, the system is able to match different surface forms of a single entry (e.g. "local area network" and "local area networks"). Note that, thanks to the deep representation used at the t-layer, we are also able to capture form variations in tokens other than the root. For instance, in Spanish both adjectives and nouns carry gender and number information, but in the t-layer only the highest node encodes this information. This way, the system will be able to match both "disco duro" ("hard disk") and "discos duros" ("hard disks") for a single dictionary entry, even if the surface form of the children node "duro" was not the same in the original text. In addition to that, it should be noted that we do allow the subtree of the text to translate to have additional children nodes to the left or right, but only at the first level below the root node, so we are able to match chunks like "corporate local area network" or "external hard disk" for the previous examples.

In order to do the matching efficiently, we use a prebuilt hash table that maps the lemma and part-of-speech pair of the root node of each dictionary entry to the full tree obtained in the preprocessing stage. This way, for each node in the input tree, we look up its lemma and part-of-speech in this hash map and, for all the occurrences, recursively check if their children nodes match.

3. **Translation:** During translation, we replace each matched subtree with the tree of its corresponding translation in the dictionary, which was built in the preprocessing stage. For that purpose, the children nodes of the matched subtree are simply removed and the ones from the dictionary are inserted in their place. As for the root node, the lemma and part-of-speech are replaced with the one from the dictionary, but all the other attributes are left unchanged. Given that these attributes are language independent, the appropriate surface form will then be generated in subsequent stages, so for our example "local area network" is translated as "red de area local" while "local area networks" is translated as "redes de area local", even if there is a single entry for them in the dictionary.

	en-cs	en-eu	en-es	en-pt
Pilot2 gazetteers	231,516	204,816	148,342	200,702
Microsoft Terminology	20,558	25,069	6,474	15,748

Table 5: Source and number of gazetteer entries in each language.

We performed experiments with the Microsoft Terminology Collection (MTC), which includes IT related terminology for several languages. We compared the performance of the system when the MTC is integrated as a gazetteer (as developed for Pilot2) or using the treelet terminology translation module. The results on Batch2a (Table 6) shows that the contribution of the treelet approach differs from language to language. For both Spanish and Basque incorporating the MTC dictionary as a gazetteer lead to a small improvement (+0.22 and +0.09) which is much bigger when the treelet approach is used (+2.15 and +2.9). The Portuguese, instead, suffers some degradation when incorporating the MTC terminology as gazetteer (-0.23), but an improvement when incorporating the same dictionary as treelets (+0.33). Finally, the Czech system has a completely different

behavior, since incorporating MTC as gazetteer leads to an improvement (+0.11), but incorporating the same dictionary using the treelet approach implies a worsening (-0.47).

	en→cs	en→es	en→eu	en→pt
TectoMT w/o gazetteers	31.45	29.61	17.15	21.96
TectoMT (Pilot2 gazetteers)	34.56	32.03	20.51	22.68
+ MTC as gazetteer	34.67	32.25	20.60	22.45
+ MTC as treelet	34.09	34.18	23.41	23.01

Table 6: Comparison of the integration of Microsoft Terminology Collection (MTC) on QTLeap batch2a testset

We performed a manual inspection of the results, in order to identify the source of the differences between languages. Evan a more detailed analysis is needed, we found that for the languages where the improvement is bigger (Spanish and Basque), there are a few very productive terms that have a big impact in the overall evaluation. Both in Spanish and Basque the English verb 'click' is translate with a non-isomorphic term (*hacer click* 'do click'). This kind of entries suppose a challenge for TectoMT, which performs isomorphic transfer at t-level. They neither can be properly handle by the gazetteer approach, since they need morphological treatment both in analysis and generation. For the rest of the languages (Czech and Portuguese), we have not found any such a productive entry and the effect of the treelet depends on many entries with few occurrences each. Such a differences denotes that the effectiveness of the approach depends on the terminology dictionary used at translation. In our case we reused a terminology collection publicly available on the internet, more suitable for some languages than for others.

2.2.1 Enriching MTC with FREME

FREME [Sasaki et al., 2015] is a framework of e-services that allows enriching end-user content in a variety of ways. One of the services it provides, e-Terminology, is used to annotate terminology.

We ran an experiment for en→pt whereby we enrich the term pairs already present in MTC with additional alternatives by searching for those terms in Freme's e-Terminology API and bringing into the terminology term pairs proposed by FREME that are new. This enrichment yielded 889 additional entries (for a total of 16,637 entries).

The enriched terminology was incorporated as treelets, but the BLEU score that we obtained (22.80) was below that achieved with MTC alone.

2.3 Summary of Experiment 5.4.4

The use of domain terminology has shown to be very useful to adapt the TectoMT system to the IT domain. The use of gazetteers allows to better translate software elements as menu items and button names. Unfortunately, the attempt to collect more terminology from comparable corpora has been mildly effective, and the new terminology collected from Wikipedia has not brought further improvements over the ones already incorporated in Pilot 2. In order to overcome some of the restrictions of the gazetteer approach, we also tested a new approach which makes use of the treelet representation of the terms and does the translation of the terms in the t-level. The treelet approach showed to be effective for some of the languages (Basque, Spanish and Portuguese) but not for Czech.

Such differences denote that the effectiveness of the approach depends on the terminology dictionary used at translation.

3 Experiments 5.4.5: New transduction algorithms

Although the MaxEnt translation models used in the previous Pilots are powerful, we decided to explore the use of an alternative model trained with VowpalWabbit [Langford et al., 2007] machine learning toolkit.[9] The VowpalWabbit model has several advantages over the original MaxEnt model:

- Only one model for all t-lemmas is trained instead of a separate model for each source t-lemma. This is technically easier to work with. It also opens space for exploring novel features shared across multiple source t-lemmas (so-called *transfer learning* using *label-dependent features*, but we have not experimented with them yet.

- The training is many times faster. Training MaxEnt t-lemma models on CzEng 1.0 takes more than one day when parallelized on 200 cores in SGE cluster (one needs to wait until the last t-lemma model is trained). Training Vowpal Wabbit t-lemma model on CzEng 1.0 takes less than two hours (with 2-pass training) on a single machine (2 cores). Both approaches require to extract the training data into a suitable format, which can be easily parallelized and takes several hours on the 200 cores cluster. It is obvious that Vowpal Wabbit allows researchers to try many more experimental setups than the MaxEnt in the same amount of time.

- No pruning of training data is needed. In order to be able to train the MaxEnt models in reasonable time, we had to limit the number of training instances per one source t-lemma to 10,000 and exclude source t-lemmas with less than 100 training instances. In VowpalWabbit no such training is needed because of the fast online learning and also because the model takes less space thanks to feature hashing.[10]

- VowpalWabbit's is trained with online learning, which allows domain adaptation using resumed learning. In our en→cs setting it means we first train two passes on CzEng and save the model. Then we take the model and continue training it with two more passes on Batch1a. Batch1a is much smaller than CzEng (1 K sentences versus 15 M sentences), but online training is more sensitive to the later training examples, so this approach is quite effective.

- The translation quality is significantly better than MaxEnt (up to +1.33 BLEU improvement on en→cs Batch3a, cf. Section 3.1).

Technically, we use cost-sensitive one-against-all reduction to logistic regression with label-dependent-features. The exact training commands are as follows:

```
$ vw -d czeng.dat.gz -f czeng.model -c --holdout_off -l 3 --passes=2 \
     --loss_function=logistic --csoaa_ldf=mc --probabilities -b 29 -qST
```

[9] The VowpalWabbit translation models in TectoMT are implemented in `T2T::EN2CS::TrLAddVariantsVW2` block (https://github.com/ufal/treex/blob/a65b6ce1/lib/Treex/Block/T2T/EN2CS/TrLAddVariantsVW2.pm) which uses VowpalWabbit from https://github.com/JohnLangford/vowpal_wabbit.

[10] Moreover, the size of the model trained with VowpalWabbit can be adapted. We use 29-bit hash function, so the models take about 3 GiB of disk and 8 GiB of memory. By using 27 bits, we could scale down the model to 2 GiB of memory with just a tiny degradation in translation quality.

```
$ vw -d batch1a.vw -f final.model -c --holdout_off -l 3 --passes=2 \
    --loss_function=logistic -i czeng.model
```

Feature space `S` contains all the source-language context features. Feature space `T` contains the conjunction of source and target t-lemma.

We have improved VowpalWabbit by implementing the option `--probabilities`, which results in outputting the whole distribution of all possible translation options and their probabilities (otherwise, VowpalWabbit reports only the most probable translation). We need this distribution because we combine the TM predictions with TreeLM scores using HMTM (cf. Deliverable D2.11). The option `--probabilities` also instructs VowpalWabbit to report the multi-class logistic loss, which we consider a better intrinsic quality indicator for our purposes than the zero-one loss which is reported by default.

Integration of VowpalWabbit (VW) is also reported in deliverable D2.11. We repeated the description of the VowpalWabbit integration here, so this deliverable is self-contained. Here we report the results of the en→cs (Section 3.1) and en→es (Section 3.2) experiments. The application to other language pairs and to formeme model is straightforward.

3.1 Results for en→cs VW in TectoMT

	Batch2a	Batch3a	Batch4a
TectoMT with MaxEnt	34.02	22.68	23.48
TectoMT with Vowpal Wabbit	34.67	24.01	24.31
improvement	+0.65	+1.33	+0.83

Table 7: Comparison of en→cs TectoMT with MaxEnt vs. Vowpal Wabbit translation model evaluated with BLEU.

Table 7 shows improvements in the en→cs Pilot3 stemming from substitution of MaxEnt with VowpalWabbit transfer models. We have not done a proper ablation analysis yet to test which components are responsible for this improvement. Possible components responsible for this improvement are:

- Using modern machine learning in VowpalWabbit instead of MaxEnt,
- No pruning of the training data,
- Online domain adaptation instead of translation models interpolation.
- Thanks to the much faster training, we were able to do a small hyper-parameter search on development data, where we found the optimal learning rate is 3 and the number of training passes 2. However, the effect of these hyper-parameters in terms of logistic loss and BLEU score was not big (doing three passes instead of two passes led only to minimal improvements, and doing one pass led to slight worsening).
- We also use a slightly enriched feature set, which considers e.g. conjunction of neighboring t-lemmas and formemes as features, while our MaxEnt considered them only separately.

In the future, we plan to investigate this and hopefully find even more effective features and learning settings.

3.2 Incorporation of semantic information with VW into en→es

As soon as the VowpalWabbit translation models were integrated in TectoMT, the new translation models were used to incorporate semantic information in en→es translation. In order to use SuperSense tags and Wornet Synsets, we reused some previous semantic tagging which does not completely share tokenization with the TectoMT Pilot. Those tokenization differences negatively impact on the translation (34.16 vs 31.47), since they interfere with some TectoMT modules such as HideIT and Gazetteers. All the experiments in this section uses the same tokenization, and are therefore comparable.

	Batch2a
TectoMT with MaxEnt	31.47
TectoMT with Vowpal Wabbit	31.54
TectoMT with VW (+Supersense IDs)	31.51
TectoMT with VW (+Synset IDs)	31.47

Table 8: Batch2a results for English–to–Spanish translations using VW transfer

Table 8 shows that the substitution of MaxEnt transfer model with VowpalWabbit model leads to a small improvement of +0.07 BLEU points. The improvement obtained in en→es is much smaller than the one obtained for en→cs, but is not clear which is the main reason. The experimentation with Spanish were carried out based on a preliminary integration of VW in TectoMT, on that time some of the features used for Czech were not available yet. Some of those features, as the `--probabilities` option, should not have a clear impact on the translation quality, and others, as the feature enrichment, should only have a limited effect. But the combination of all of them, as well as the differences between languages and training data characteristics, results in such a big difference between languages.

The incorporation of semantic information using the new VW transfer models does not implies any improvement, and the results suffers a small degradation in comparison with the original Vowpal Wabbit transfer models when incorporating SuperSense IDs (-0.03) or Wordnet's Synset IDs (-0.07) as extra features. This results are consistent with the results in Section 4.1, where the same kind of information is integrated on the MaxEnt translation models.

3.3 Summary of Experiment 5.4.5

At the time of preparing the TectoMT system for Pilot 3, the VowpalWabbit experiments had not yielded any significant improvement, and it was thus discarded. The same decision was taken respect of the of VowpalWabbit to profit from the Supersense and Synset tags. The positive results for using VowpalWabbit here reported for Czech came just in time to be included in Pilot 3 for Czech, but too late for the rest of the languages.

4 Additional experiments

In this section we include the results of the experiments which have been carried over from the second year to new languages (enriching word representations, improvement of WSD and MT for Bulgarian), an analysis of potential effects of applying coreference when translating into Spanish and Basque, as well as improvements on the quality of Named Entity Disambiguation for English.

4.1 Experiments 5.4.2: Enriching word representations

The goal of Experiment 5.4.2 was to improve upon the experiment 5.4.1 by enriching word (and lemma) representations with concept information coming from Word Sense Disambiguation (WSD) software. In this round, we wanted to check whether the improvements for Portuguese reported in D5.7 carry out to other languages. More precisely, the incorporation of domain-adapted WSD information was helpful when the en→pt Translation Model (TM) was trained on a big general domain corpus, as shown in Table 9, carried over from D5.7.

Method	Node	+Parent	+Siblings	All
Baseline		18.31		
Synset IDs	18.43*	18.45*	18.46*	18.35
Supersense IDs	18.44*	18.30	18.44*	18.46*
Both	18.34	18.50*	18.41*	18.37

Table 9: The BLEU scores (Batch2a) of the en→pt translation using WordNet information as features in the lemma-to-lemma Discriminative TM with a domain-adapted WSD on the Europarl data. The symbol * denotes statistically significant ($p < 0.05$) improvement compared to the baseline.

According to these results, a TM trained on a general domain corpus (Europarl) can be successfully enriched with WSD information to significantly improve the translation. Due to this positive results, we decided to enrich the TM of the other language pairs the same way. So, we applied the same domain-adapted WSD system to the English part of our bilingual training corpora, and trained a set of TM which incorporated synset IDs and SuperSense tags as additional features in the lemma-to-lemma Discriminative TM. The features incorporated include the synset ID and/or the SuperSense tag of the node that has to be translated, as well as the semantic information of the parent and the siblings words. The best configuration for Portuguese is the combination of the synset ID and SuperSense tag of both the node and his parent. Nevertheless, the differences between some configurations are not big, and we decided to test all of them for the new language pairs.

Tables 10–12 show the results for Spanish, Basque and Czech. Unfortunately, none of the configurations tried for Portuguese worked for these languages.

4.1.1 Summary of Experiment 5.4.2

Including sense information as features of the Discriminative TM of TectoMT was successfully tested on en→pt, but the positive result does not carry over to other language pairs. The explanation for this failure could lay in the fact that WSD information is

	Node	+Parent	+Siblings
Baseline		34.16	
Synset IDs	33.99	33.81	33.84
Supersense IDs	34.12	33.85	33.97
Both	34.08	33.82	33.96

Table 10: The BLEU scores (Batch2a) of the en→es translation using WordNet information as features in the lemma-to-lemma Discriminative TM.

	Node	+Parent	+Siblings
Baseline		23.41	
Synset IDs	23.41	23.35	23.34
Supersense IDs	23.36	23.30	23.30
Both	23.32	23.31	23.30

Table 11: The BLEU scores (Batch2a) of the en→eu translation using WordNet information as features in the lemma-to-lemma MaxEnt TM.

	Node	+Parent	+Siblings
Baseline		34.56	
Synset IDs	34.18	33.92	33.93
Supersense IDs	34.42	33.91	34.01
Both	34.35	33.88	33.99

Table 12: The BLEU scores (Batch2a) of the en→cs translation using WordNet information as features in the lemma-to-lemma Discriminative TM.

too weak to drive the TectoMT system towards different translations. Alternatively, the specific target domain for the MT engine, PCMEDIC questions and answers, might not be the most suitable for profiting from WSD information. In fact, Portuguese TM are trained on Europarl, while both ES, EU and CS models use interpolation of in-domain and out-of-domain corpora.

4.2 Improving English to Bulgarian MT with WSD

Bulgarian ↔ English Pilot 3 systems are implemented as a hybrid machine translation system consisting of three main steps (depicted in Figure 1). The source-language text is linguistically annotated, then translated with the Moses system to the target language and post-processed using the linguistic annotation projected from the source side to the target side.

During the translation with the Moses system the word alignment is stored in order to be used for the projection of the linguistic analyses from the source text to the target text.

It is important to mention that the number of the tokens in the source and the target language might differ. Also, the alignments can include many-to-many correspondences, not just one-to-one. Nevertheless, in practice about 80 % of the alignments are one-to-one or two-to-two tokens.

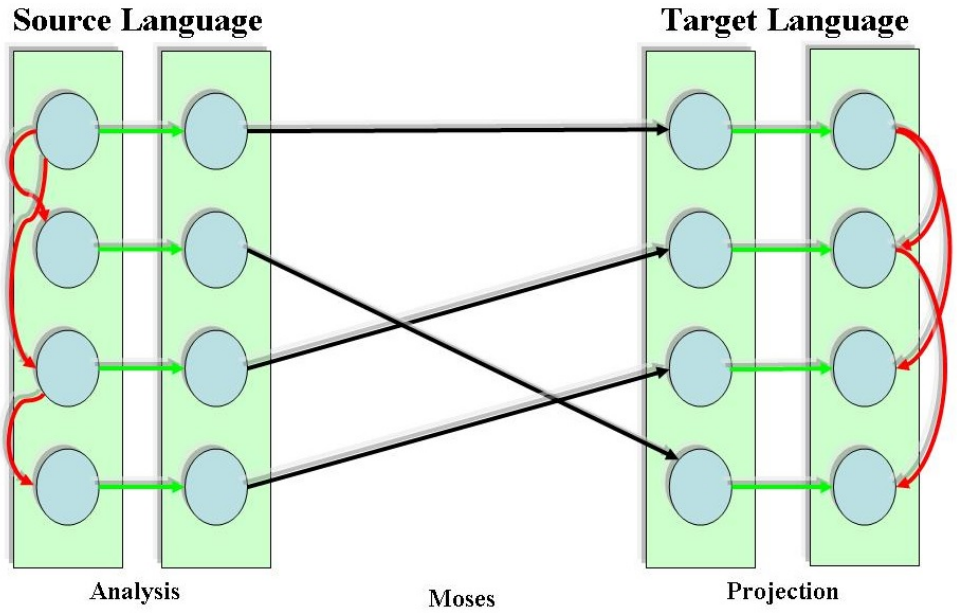


Figure 1: A hybrid architecture of bg↔en Pilot 3 for transferring linguistic information from the source to the target language. The linguistic analyses for the source language (Analysis - column 1) are projected to a tokenized source text (Analysis - column 2); then the Moses models (Moses) are applied for producing a target language translation. The translation alignment (Projection - column 1) is used for transferring the information to the corresponding tokens in the target language (Projection - column 2). The projected linguistic information interacts with the linguistic features of the tokens in the target text (for example the morphosyntactic features). Finally, the resulting annotation of the target text is used for post-processing.

For the English → Bulgarian Pilot 3 system we use the factor-based statistical machine translation developed for Pilot2 (see Deliverable 5.7). The main idea behind this model was to use WSD for the construction of an intermediate representation of the source text, called source/target (S/T) text. This intermediate text is the source text linguistically annotated with POS tags and WordNet synsets produced by WSD system UKB[11]. The analysis of English (tokenization, lemmatization, POS tagging and depen-

[11] http://ixa2.si.ehu.es/ukb/

dency parsing) as a source language was done with the CoreNLP tools[12], developed at Stanford University. On the basis of aligned English to Bulgarian WordNet some of the input English word forms are substituted by Bulgarian lemmas, where available. Here we present the improvement of this factor-based model. The S/T factor-based Moses model (Pilot-3WSD) was incorporated in the overall architecture for en→bg translation model described in Deliverable D2.11.

The motivation for using the representative lemma in the target language was our expectation that the various synset IDs would unify with the similar translations in the target language. For example, in the en→bg direction, the two concepts referred by *donor*: `wn30-10025730-n` ("person who makes a gift of property") and `wn30-10026058-n` ("a medical term denoting someone who gives blood or tissue or an organ to be used in another person") are very close to each other. They have the same translation in Bulgarian in both corresponding synsets: *донор*. The representative word is selected on the basis of a frequency list of Bulgarian lemmas constructed over large corpora (70 million words).

Here is an illustration of the procedure that was performed with respect to the training, testing and tuning of the Moses system:

English sentence:
This is real progress .

English sentence with factors:
this|this|dt is|be|vbz реален|real|jj напредък|progress|nn .|.|.

Bulgarian sentence with factors:
това|това|pd е|съм|vx реален|реален|a напредък|напредък|nc .|.|pu

Bulgarian sentence:
Това е реален напредък.

The idea was to enrich S/T text with target-language factors. The experiments we performed during the third year using S/T text representation showed improvement for the translation, so we selected such a Moses model for en→bg. For the direction bg→en the result did not improve and for that reason we do not report it here.

Table 13 presents the results for all pilots tested on Batch4a. The main improvement between Pilot-2 and Pilot-3WSD has come from the better WSD. In order to improve WSD we had enriched the knowledge graph with new relations extracted from sense annotated corpora such as XWN [Mihalcea and Moldovan, 2001] and SemCor [Miller et al., 1993]. The procedure for the extraction of the new relations for the knowledge graphs is described in detail by Simov et al. [2016]. The new relations include relations extracted from the logical form representation of the glosses in XWN; from the sentences annotated with WordNet synsets from XWN glosses; and from the SemCor semantic annotation.

Although Pilot-3WSD has not improved the result of Pilot-0, it remained very close to it. One reason for this fact might be the need of smart combination among the pieces of linguistic information, whose best balance has not been found yet.

However, the Pilot-3WSD model improved significantly over the same model in Pilot-2. This result shows that the better handling of the WSD improves the machine translation quality.

[12] http://stanfordnlp.github.io/CoreNLP/

System	factors	BLEU
Pilot-0	SWF	20.3
Pilot-1	SWF, SLM, MSRF	18.6
Pilot-2	TLM|SWF, SLM, SPOS	16.42
Pilot-3WSD	TLM|SWF, SLM, SPOS	19.79

Table 13: BLEU was measured on the translation of Batch4a with Pilot-0, Pilot-1, Pilot-2 and Pilot-3WSD models. The baseline is Pilot-0 which is a phrase-based model that uses only source language word forms (SWF), Pilot-1 is a factor-based Moses model that uses source language word forms (SWF), source language lemmas (SLM) and factors extracted from the Minimal Recursion Semantics (MSRF) representation. Pilot-2 and Pilot-3WSD are factor-based Moses models that use the target language lemma or the source language word form (TLM|SWF), the source language lemma (SLM) and the source language POS tags as factors. The target language lemma is selected on the basis of the WSD over the source language.

4.2.1 Summary of English to Bulgarian WSD experiment

The results obtained for English to Bulgarian translation show that an improved algorithm for WSD is effective for this pair of languages and machine translation architecture, showing promise for other languages and architectures.

4.3 Analysis of coreference for Basque and Spanish

In D5.7 we showed that resolving English coreference improved translation quality when translating to Czech and Dutch. Following this line, we performed a linguistic analysis of the possible benefits of implementing similar modifications to the systems translating from English to Basque and Spanish. We analyzed the anaphoric pronouns that were productive for Czech and Dutch, namely, personal pronouns, demonstrative pronouns, reflexive pronouns and relative pronouns. After analyzing the first 50 answers from the QTLeap batch2 for the above mentioned anaphoric pronouns we found that 35 of the answers included such pronouns, 59 instances in total (see Table 14).

Item	Frequency
personal pronoun – you	27
personal pronoun - it	9
personal pronoun – them	1
possessive pronoun – your	2
(implicit) relative pronoun – that, (prep) what, (prep) which	17
quantitative pronoun – one	3

Table 14: Frequencies of anaphoric pronouns in a sample of 50 answers from QTLeap batch2.

We analyzed how the English-to-Spanish and the English-to-Basque systems resolved such cases without any coreference information (see Tables 15 and 16). We noticed that in the case of Spanish, number and gender emerged as translation issues. Number was an issue in the case of the pronoun *you*, which can refer to either the second person singular or plural, informal or polite. However, given the translation domain, the system was set to output the second person singular in its polite form, which is the correct translation in this domain. The third person pronoun *them* did cause issues for the system as information about gender was necessary to resolve it properly. However, the system output the masculine option by default, which resulted in three out of four cases to be correct. Issues with relative pronouns were avoided by having the system use the gender- and number-neutral pronoun *que* rather than gender and number specific *el cual, la cual, los cuales, las cuales.*

Item	Ambiguity in Spanish	Spanish Translation
personal pronoun – you	2p pl/sg, f/inf	fixed: 2p sg f - usted
personal pronoun - it	masc/fem	subject: omit; object: gender: lo/la
personal pronoun – them	masc/fem	when object: gender – los las
possessive pronoun – your	2p pl/sg, f/in	fixed: 2p sg, f - su
relative pronoun	pronoun cual or que	fixed: que
implicit relative pronoun	pronoun cual or que	fixed: que
quantitative pronoun – one	masc/fem	gender uno/una

Table 15: Types of anaphoric elements studied in QTLeap batch2, with their ambiguity and system output for Spanish.

In the case of Basque, the anaphoric elements did not show any ambiguity, that is, the lack of specificity showed by the English source text could be naturally transferred to Basque. The English pronouns can be matched to fixed Basque pronouns which do not need further person/number/gender information to be translated correctly.

Item	Ambiguity in Basque	Basque Traslation
personal pronoun – you	same	fixed: zu
personal pronoun - it	same	fixed: hura
personal pronoun – them	same	fixed: haiek
possessive pronoun – your	same	fixed: zure
relative pronoun	same	fixed: suffix -n
quantitative pronoun – one	same	fixed: bat

Table 16: Types of anaphoric elements studied in QTLeap batch2, with their ambiguity and system output for Basque.

We also analyzed the first 50 sentences from News corpus for the aforementioned anaphoric pronouns. Again, we found that 35 sentences contained the studied pronouns, 77 in total (see Table 17).

Item	Frequency
personal pronoun – you, he, it, them	31
possessive pronoun – your, their, his	14
demonstrative pronoun – this, these/those	5
quantitative pronoun – few, some, many, much, one, ones, no one	13
quantitative reflexive pronoun – oneself	1
(implicit) relative pronoun – that	5
relative pronoun – what, which, who, when	8

Table 17: Frequencies of anaphoric pronouns in a sample of 50 sentences in the QTLeap news corpus.

As can be seen, the variety of pronouns was much wider in this set. However, the results for Spanish and Basque were very much in line with the previous analysis (see Tables 18 and 19). For Spanish, number and gender issues posed a problem but were mostly well resolved by using pre-established decisions on number and formality, by using neutral options and by having default masculine translations. This left only a small number of incorrectly translated cases. For Basque, the ambiguity could be maintained in the target language with no decrease in quality (see Table 19).

The low margin for improvement discovered during the analysis showed that solving coreference in English would not improve the quality of translation for Basque and Spanish on these domains.

4.3.1 Summary of coreference experiments

The analysis of the translation of coreferents from English to Basque and Spanish showed that solving the coreference for English and then porting that information to Spanish and Basque would not improve the quality of translation, as the coreferent in English could be translated correctly in most of the cases without the need of that information.

Item	Ambiguity	Spanish Translation
pers. pronoun – you	2p sg/pl, f/inf	fixed number: 2p sg, no fixed formality
pers. pronoun – it	masc/fem	subject: omit; object: gender: lo/la
pers. pronoun – he	same	omitted (13/14)
pers. pronoun – them	masc/fem	object: gender: los/las/les
poss. pronoun – your	2p sg/pl, f/inf	fixed: 2p sg, no fixed formality
poss. pronoun – their	masc/fem	fixed: su
poss. pronoun – his	fame	fixed: su
dem. pronoun – this, these/those	masc/fem	gender: ése/éste/-a/-o
quant. pronoun – few, some, many, much...	masc/fem	gender: tanto/alguno/.../-a/-os/-as
quantitative reflexive pronoun – oneself	masc/fem	gender: uno/una
(implicit) relative pronoun – that	pronoun cual or que	fixed: que
relative pronoun – what	same	fixed: lo que
relative pronoun – which	pronoun cual or que	fixed: que
relative pronoun – who	sg/pl	fixed: que
relative pronoun – when	sg/pl, masc/fem	gender: en el/la/los/las que
implicit relative pronoun	pronoun cual or que	fixed: que
quantitative pronoun – one	masc/fem	gender: uno/una

Table 18: Types of anaphoric elements studied in QTLeap new corpus, with their ambiguity and system output for Spanish.

Item	Ambiguity	Basque Translation
pers. pronoun – you	pl or sing	zu / zuek
pers. pronoun – he, it, them	same	fixed: empty, hura, haiek, omit
poss. pronoun – your, their, his	same	fixed: zure / haien / bere
dem. pronoun – this, these/those	same	fixed: hau / hauek
quant. pron. – few, some, many, much...	same	fixed: batzuk, asko, inor...
quantitative reflexive pronoun – oneself	same	fixed: norbera
(implicit) relative pronoun – that	same	fixed: suffix -n (rel behind)
relative pronoun – what, which, who, when	same	fixed: suffix -n (rel behind)

Table 19: Types of anaphoric elements studied in QTLeap new corpus, with their ambiguity and system output for Basque.

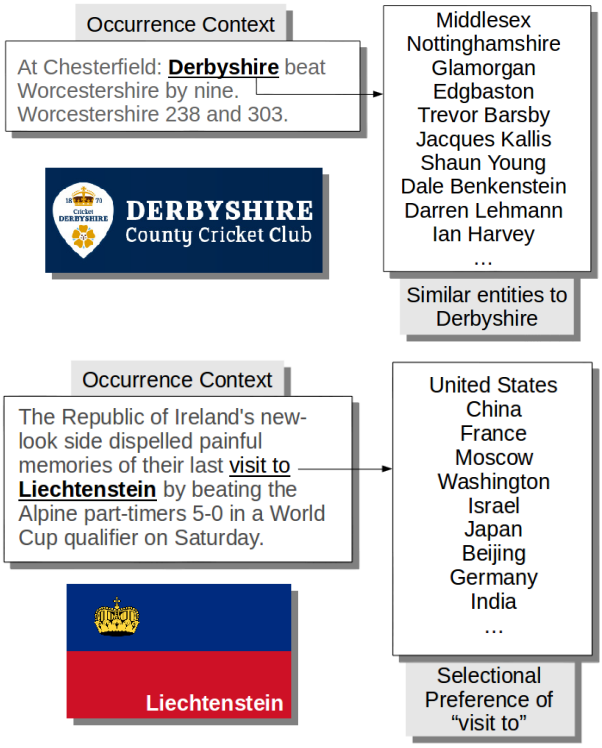


Figure 2: Two examples where NED systems fail, motivating our two background models: similar entities (top) and selectional preferences (bottom). The logos correspond to the gold label.

4.4 Improving Named Entity Disambiguation for English

The workplan for the WP5 working package includes research on methods to improve the advanced processors until the end of the project. Deliverable D5.9 reported the status of the tools used to produce the resources in D5.8. Section 10 of D5.9 reported some further improvements on Word Sense Disambiguation (Bulgarian, Czech, English and Spanish) and Named Entity Recognition and Classification (Basque, English and Spanish).

In this section we report further improvements on NED for English. More specifically, UPV/EHU has explored methods to overcome poor and misleading contexts, a problem which hurts NED performance. Below we describe experiments which show that we can alleviate the problem, thanks to the the acquisition of two kinds of background information: entity similarity and selectional preferences for syntactic positions. We show, using a generative Näive Bayes model for NED Barrena et al. [2015], that the additional sources of context are complementary, and improve results.

Motivation According to Wikipedia, *Liechtenstein* can refer to the micro-state, several towns, two castles or a national football team, among other instances. Another ambiguous entity is *Derbyshire* which can refer to a county in England or a cricket team. Most NED research use knowledge-bases derived or closely related to Wikipedia.

Figure 2 shows two real examples from the development dataset which contains text from News, where the clues in the context are too weak or misleading. In fact, two mentions in those examples (*Derbyshire* in the first and *Liechtenstein* in the second) are wrongly disambiguated by a bag-of-words context model.

In the first example, the context is very poor, and the system returns the *county* in-

stead of the *cricket team*. In order to disambiguate it correctly one needs to be aware that *Derbyshire*, when occurring on News, is most notably associated with cricket. This background information can be acquired from large News corpora such as Reuters [Lewis et al., 2004], using distributional methods to construct a list of closely associated entities [Mikolov et al., 2013]. Figure 2 shows entities which are distributionally similar to *Derbyshire*, ordered by similarity strength. Although the list might say nothing to someone not acquainted with cricket, all entities in the list are strongly related to cricket: Middlesex used to be a county in the UK that gives name to a cricket club, Nottinghamshire is a county hosting two powerful cricket and football teams, Edgbaston is a suburban area and a cricket ground, the most notable team to carry the name Glamorgan is Glamorgan County Cricket Club, Trevor Barsby is a cricketer, as are all other people in the distributional context. When using these similar entities as context, our system does return the correct entity for this mention.

In the second example, the words in the context lead the model to return the *football team* for *Liechtenstein*, instead of the *country*, without being aware that the nominal event "visit to" prefers locations arguments. This kind of background information, known as selections preferences, can be easily acquired from corpora [Erk, 2007]. Figure 2 shows the most frequent entities found as arguments of "visit to" in the Reuters corpus. When using these filler entities as context, the context model does return the correct entity for this mention.

Acquiring background information We built our two background information resources from the Reuters corpus [Lewis et al., 2004], which comprises 250K documents. We chose this corpus because it is the one used to select the documents annotated in one of our gold standards (cf. Section 4.4). The documents in this corpus are tagged with categories, which we used to explore the influence of domains.

The documents were processed using a publicly available NLP pipeline, Ixa-pipes,[13] including tokenization, lematization, dependency tagging and NERC.

Similar entity mentions: Distributional similarity is known to provide useful information regarding words that have similar co-occurrences. We used the popular word2vec[14] tool to produce vector representations for named entities in the Reuters corpus. In order to build a resource that yields similar entity mentions, we took all entity-mentions detected by the NERC tool and, if they were multi word entities, joined them into a single token replacing spaces with underscores, and appended a tag to each of them. We run word2vec with default parameters on the pre-processed corpus. We only keep the vectors for named entities, but note that the corpus contains both named entities and other words, as they are needed to properly model co-occurrences.

Given a named entity mention, we are thus able to retrieve the named entity mentions which are most similar in the distributional vector space. All in all, we built vectors for 95K named entity mentions. Figure 2 shows the ten most similar named entities for *Derbyshire* according to the vectors learned from the Reuters corpus. These similar mentions can be seen as a way to encode some notion of a topic-related most frequent sense prior.

Selectional Preferences: Selectional preferences model the intuition that arguments of predicates impose semantic constraints (or preferences) on the possible fillers for that argument position [Resnik, 1996]. In this work, we use the simplest model, where the

[13] http://ixa2.si.ehu.es/ixa-pipes/

[14] https://code.google.com/archive/p/word2vec/

selectional preference for an argument position is given by the frequency-weighted list of fillers [Erk, 2007].

We extract dependency patterns as follows. After we parse Reuters with the Mate dependency parser [Bohnet, 2010] integrated in IxaPipes, we extract $(H \xrightarrow{D} C)$ dependency triples, where D is one of the Subject, Object or Modifier dependencies[15] (SBJ, OBJ, MOD, respectively), H is the head word and C the dependent word. We extract fillers in both directions, that is, the set of fillers in the dependent position $\{C : (H \xrightarrow{D} C)\}$, but also the fillers in the head position $\{H : (H \xrightarrow{D} C)\}$. Each such configuration forms a template, $(H \xrightarrow{D} *)$ and $(* \xrightarrow{D} C)$.

In addition to triples (single dependency relations) we also extracted tuples involving two dependency relations in two flavors: $(H \xrightarrow{D_1} C_1 \xrightarrow{D_2} C_2)$ and $(C_1 \xleftarrow{D_1} H \xrightarrow{D_2} C_2)$. Templates and fillers are defined as done for single dependencies, but, in this case, we extract fillers in any of the three positions and we thus have three different templates for each flavor.

As dependency parsers work at the word level, we had to post-process the output to identify whether the word involved in the dependency was part of a named entity identified by the NERC algorithm. We only keep tuples which involve at least one name entity. Some examples for the three kinds of tuples follow, including the frequency of occurrence, with entities shown in bold:

(beat $\xrightarrow{SBJ}$ **Australia**) 141
(refugee $\xrightarrow{MOD}$ **Hutu**) 1681
(visit $\xrightarrow{MOD}$ to $\xrightarrow{MOD}$ **United States**) 257
(match $\xrightarrow{MOD}$ against $\xrightarrow{MOD}$ **Manchester United**) 12
(Spokesman $\xleftarrow{SBJ}$ tell $\xrightarrow{OBJ}$ **Reuters**) 1378
(**The Middle East** $\xleftarrow{MOD}$ process $\xrightarrow{MOD}$ peace) 1126

When disambiguating a mention of a named entity, we check whether the mention occurs on a known dependency template, and we extract the most frequent fillers of that dependency template. For instance, the bottom example in Figure 2 shows how *Liechtenstein* occurs as a filler of the template (visit $\xrightarrow{MOD}$ to $\xrightarrow{MOD}$ *), and we thus extract the selectional preference for this template, which includes, in the figure 2, the ten most frequent filler entities.

We extracted more than 4.3M unique tuples from Reuters, producing 2M templates and their respective fillers. The most frequent dependency was MOD, followed by SUBJ and OBJ [16] The selectional preferences include 400K different named entities as fillers.

Note that selectional preferences are different from dependency path features. Dependency path features refer to features in the immediate context of the entity mention, and are sometimes added as additional features of supervised classifiers. Selectional preferences are learnt collecting fillers in the same dependency path, but the fillers occur elsewhere in the corpus.

NED system: The disambiguation system is a Näive Bayes model as initially introduced by Han and Sun [2011], but adapted to integrate the background information

[15] Labels are taken from the Penn Treebank https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html

[16] 1.5M, 0.8M and 0.7M respectively

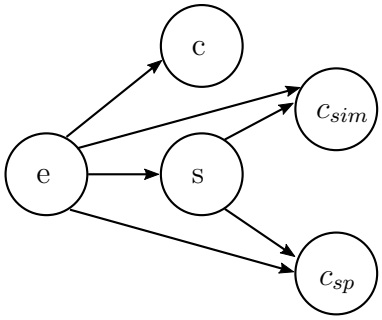


Figure 3: Dependencies among variables in our Bayesian network.

extracted from the Reuters corpus. The model is trained using Wikipedia,[17] which is also used to generate the entity candidates for each mention.

Following usual practice, candidate generation is performed off-line by constructing an association between strings and Wikipedia articles, which we call dictionary. The association is performed using article titles, redirections, disambiguation pages, and textual anchors. Each association is scored with the number of times the string was used to refer to the article [Agirre et al., 2015]. We also use Wikipedia to extract training mention contexts for all possible candidate entities. Mention contexts for an entity are built by collecting a window of 50 words surrounding any hyper link pointing to that entity.

Both training and test instances are pre-processed the same way: occurrence context is tokenized, multi-words occurring in the dictionary are collapsed as a single token (longest matches are preferred). All occurrences of the same target mention in a document are disambiguated collectively, as we merge all contexts of the multiple mentions into one, following the one-entity-per-discourse hypothesis [Barrena et al., 2014].

The Näive Bayes model is depicted in Figure 3. The candidate entity e of a given mention s, which occurs within a context c, is selected according to the following formula:

$$e = \arg\max_e P(s, c, c_{\text{sp}}, c_{\text{sim}}, e) =$$
$$\arg\max_e P(e)P(s|e)P(c|e)P(c_{\text{sp}}|e, s)P(c_{\text{sim}}|e, s)$$

The formula combines evidences taken from five different probabilities: the entity prior $p(e)$, the mention probability $p(s|e)$, the textual context $p(c|s)$, the selectional preferences $P(c_{\text{sp}}|e, s)$ and the distributional similarity $P(c_{\text{sim}}|e, s)$. This formula is also referred to as the "**Full model**", as we also report results of partial models which use different combinations of the five probability estimations.

Entity prior $P(e)$ represents the popularity of entity e, and is estimated as follows:

$$P(e) \propto \frac{f(*, e) + 1}{f(*, *) + N}$$

where $f(*, e)$ is the number of times the entity e is referenced within Wikipedia, $f(*, *)$ is the total number of entity mentions and N is the number of distinct entities in Wikipedia. The estimation is smoothed using the *add-one* method.

Mention probability $P(s|e)$ represents the probability of generating the mention s given the entity e, and is estimated as follows:

[17] We used a dump from 25-5-2011. This dump is close in time to annotations of the datasets used in the evaluation (c.f. Section 4.4)

$$P(s|e) \propto \theta \frac{f(s,e)}{f(*,e)} + (1-\theta)\frac{f(s,*)}{f(*,*)}$$

where $f(s,e)$ is the number of times mention s is used to refer to entity e and $f(s,*)$ is the number of times mention s is used as anchor. We set the θ hyper-parameter to 0.9 according to developments experiments in the CoNLL testa dataset.

Textual context $P(c|e)$ is the probability of entity e generating the context $c = \{w_1, \ldots, w_n\}$, and is expressed as:

$$P(c|e) = \prod_{w \in c} P(w|e)^{\frac{1}{n}}$$

where $\frac{1}{n}$ is a correcting factor that compensates the effect of larger contexts having smaller probabilities. $P(w|e)$, the probability of entity e generating word w, is estimated following a bag-of-words approach:

$$P(w|e) \propto \lambda \frac{c(w,e)}{c(*,e)} + (1-\lambda)\frac{f(w,*)}{f(*,*)}$$

where $c(w,e)$ is the number of times word w appears in the mention contexts of entity e, and $c(*,e)$ is the total number of words in the mention contexts. The term in the right is a smoothing term, calculated as the likelihood of word w being used as an anchor in Wikipedia. λ is set to 0.9 according to development experiments done in CoNLL testa.

Distributional Similarity $P(c_{\text{sim}}|e,s)$ is the probability of generating a set of similar entity mentions given an entity mention pair. This probability is calculated and estimated in exactly the same way as the textual context above, but replacing the mention context c with the mentions of the 30 most similar entities for s instead.

Selectional Preferences $P(c_{\text{sp}}|e,s)$ is the probability of generating a set of fillers c_{sp} given an entity and mention pair. The probability is again analogous to the previous ones, but using the filler entities of the selectional preferences of s instead of the context c. In our experiments, we select the 30 most frequent fillers for each selectional preferences, concatenating the filler list when more than one selectional preference is applied.

Ensemble model: In addition to the Full model, we created an ensemble system that combines the probabilities described above using a weighting schema, which we call "**Full weighted model**". In particular, we add an exponent coefficient to the probabilities, thus allowing to control the contribution of each model.

$$\arg\max_{e} P(e)^{\alpha} P(s|e)^{\beta} P(c|e)^{\gamma} P(c_{\text{sp}}|e,s)^{\delta} P(c_{\text{sim}}|e,s)^{\omega}$$

We performed an exhaustive grid search in the interval $(0,1)$ for each of the weights, using a step size of 0.05, and discarding the combinations whose sum is not one. Evaluation of each combination was performed in the CoNLL testa development set, and the best combination was applied in the test sets.[18]

[18] The best combination was $\alpha = 0.05$, $\beta = 0.1$, $\gamma = 0.55$ $\delta = 0.15$, $\omega = 0.15$

Dataset	Documents	Mentions
CoNLL testa	216	4791
CoNLL testb	231	4485
TAC2014 DEL test	138	2817

Table 20: Document and linkable mention counts for CoNLL and TAC2014 DEL datasets.

System	CoNLL	TAC14
$P(e)P(s\|e)$	73.07	78.31
$P(e)P(s\|e)P(c\|e)$	79.98	82.11
$P(e)P(s\|e)P(c\|e)P(c_{\text{sp}}\|e,s)$	81.31	82.61
$P(e)P(s\|e)P(c\|e)P(c_{\text{sim}}\|e,s)$	82.72	83.24
Full	82.85	83.21
$P(e)^{\alpha}P(s\|e)^{\beta}P(c\|e)^{\gamma}$	86.44	81.61
Full weighted	**88.32**	**83.46**

Table 21: Overall micro accuracy results on the CoNLL testb and TAC 2014 DEL datasets.

Evaluation Datasets: The evaluation has been performed on one of the most popular datasets, the CoNLL 2003 named-entity disambiguation dataset, also know as the AIDA or CoNLL-Yago dataset [Hoffart et al., 2011]. It is composed of 1393 news documents from Reuters Corpora where named entity mentions have been manually identified. It is divided in three main parts: *train*, *testa* and *testb*. We used *testa* for development experiments, and *testb* for the final results and comparison with the state-of-the-art. We ignored the training part.

In addition, we also report results in the Text Analysis Conference 2014 Diagnostic Entity Linking task dataset (TAC DEL 2014).[19] The gold standard for this task is very similar to the CoNLL dataset, where target named entity mentions have been detected by hand. Through the beginning of the task (2009 to 2013) the TAC datasets were query-driven, that is, the input included a document and a challenging and sometimes partial target-mention to disambiguate. As this task also involved mention detection and our techniques are sensitive to mention detection errors, we preferred to factor out that variation and focus on the 2014.

The evaluation measure used in this experiment is micro-accuracy, that is, the percentage of linkable mentions that the system disambiguates correctly, as widely used in the CoNLL dataset. Note that TAC2014 EDL included several evaluation measures, including the aforementioned micro-accuracy of linkable mentions, but the official evaluation measure was Bcubed+ F1 score, involving also detection and clustering of mentions which refer to entities not in the target knowledge base. We decided to use the same evaluation measure for both datasets, for easier comparison. Table 20 summarizes the statistics of the datasets used in this experiment where document and mention counts are presented.

Results: We report the result of our model in the popular CoNLL testb and TAC2014 DEL datasets, which allow to compare to the state-of-the-art in NED.

Table 21 reports our results, confirming that both background information resources improve the results over the standard NED generative system, separately, and in combination, for both datasets (Full row). All differences with respect to the standard generative

[19] http://www.nist.gov/tac/2014/KBP/

system are statistically significant according to the Wilcoxon test (p-value < 0.05).

In addition, we checked the contribution of learning the ensemble weights on the development dataset (testa). Both the generative system with and without background information improve considerably.

The error reduction between the weighted model using background information (Full weighted row) and the generative system without background information (previous row) exceeds 10% in both datasets, providing very strong results, and confirming that the improvement due to background information is consistent across both datasets, even when applied on a very strong system. The difference is statistically significant in both datasets.

4.4.1 Summary of improvements for NED

Our experiments show that it is possible to improve the results of a state-of-the-art NED system [Barrena et al., 2015], thanks to the additional information on selectional preferences and entity similarity learned from un-annotated corpora.

5 Results of lexical semantics on Pilot 3 systems

In this section, we describe the results of the techniques that were successful in the experiments mentioned above. Table 22 summarizes the experiments presented in this deliverable, specifying which ones were successful and which ones have been integrated in Pilot 3 for which language pairs.

Exper	Sec.	MT	LS technique	Languages	Datasets	OK	P3
5.4.4	2.1	TectoMT	Wikitailor	en↔es	QTa,QTq		
	2.2	TectoMT	Treelets	en↔{cs,es,eu,pt}	QTa,QTq	es,eu,pt	es,eu,pt
5.4.5	3	TectoMT	VowpalWabbit	en→{cs,es}	QTa	cs	cs
5.4.2	4.1	TectoMT	WSD (UKB)	en↔{cs,es,eu,pt}	QTa	pt	pt
	4.2	DFMT	WSD (UKB)	en→bg	QTa	bg	bg
Coref	4.3	TectoMT	Coreference	en→{es,eu}	QTa		

Table 22: Summary of experiments, including success and integration in Pilot 3. Columns stand for the following. **MT** for the MT platform, TectoMT. **Datasets**: QTa for QTLeap answers, QTq for QTLeap queries. **OK** for successful improvement over baseline. **P3** for use in Pilot 3.

The successful components employing lexical semantics that were introduced in the previous sections are evaluated within the Pilot 3 systems. The results in terms of BLEU scores are shown in Tables 23 and 24 for translation to English and from English, respectively, except the experiments for Bulgarian, which were reported in Section 4.2. All the systems were evaluated on the Batch 2 dataset, which is used as a development dataset. The tables report several baselines, including the scores of Pilots 0,[20] 1 and 2. The row denoted as Pilot3-minus-LS shows BLEU scores of the Pilot 3 systems where all the lexical semantics components are switched off. The rows below present the effect of switching on each of the lexical semantics components relative to the Pilot3-minus-LS system. Note that there are several blank cells, which correspond to the following cases: treelets could not be applied for pt→en because of software incompatibilities; treelets were not applied to Dutch because this language was not originally in WP5; VowpalWabbit was only applied to en→cs because it only showed positive results in the final stage of the project.

The "Δ total LS" row shows the effect of switching on all the components with positive deltas. Note that Pilot 3 did not activate all components (rows) reported in the table. The components which significantly hurt performance were the use of a gazetteer in nl→en and Terminology Treelets in cs→en and en→eu. Furthermore, the use of the WSD components (+synset and +synset&supersense) in pt→en has shown to perform worst when paired with the other components and were thus not include in the Pilot 3. The final performance of the full Pilot 3 systems can be found in the last row of the table.

The "Δ total LS" is usually not a sum of the deltas for individual components, as the effects of these components may overlap. Moreover, some of these overlaps are systematic: by activating the VowpalWabbit transfer model (with online domain adaptation integrated as described in Section 3), we deactivate the domain adaptation using TM interpolation. Thus, "Δ VowpalWabbit" cannot be summed with "Δ adaptation by TM interpolation".

[20] The Pilot 0 results for en→es and es→en reported here are Pilot 0-comparable, that is Pilot 0 trained on Europarl only, so it can be fairly compared with Pilots 1, 2 and 3, which are also trained on Europarl only.

All in all, the tables show that lexical semantic techniques are largely beneficial, with positive improvements in all languages, up to 3.15 points for translation into English, and up to 11.77 for translations from English. In particular, the most successful techniques are adaptation by TM interpolation and gazeteers. The Treelets introduced in the last year are beneficial for Spanish, Portuguese and Basque (when translating from English), and VowpalWabbit is beneficial for Czech.

system	cs→en	es→en	eu→en	nl→en	pt→en
Pilot0	26.44	39.30	25.29	36.45	22.59
Pilot1	26.81	16.05	4.75	34.46	10.14
Pilot2	30.28	27.22	14.07	45.93	13.51
Pilot3-minus-LS	29.02	24.85	14.12	44.46	12.77
Δ "fixed" entities (HideIT)	+0.00	+0.00	+0.06	+0.00	+0.02
Δ specialized lexicons (gazetteers)	+0.89	+0.59	+0.00	−0.34	+0.03
Δ adaptation by TM interpolation	+1.14	+1.09	+1.48	+1.85	+1.66
Δ terminology treelets	−1.13	+2.03	−0.04		
Δ total LS	+2.12	+3.15	+1.44	+1.85	+1.71
full Pilot3	31.14	28.00	15.56	46.31	14.48

Table 23: Translations to English (Batch2q). Effect of various lexical semantic modules on BLEU performance.

system	en→cs	en→es	en→eu	en→nl	en→pt
Pilot0	31.07	25.11	28.37	32.94	19.36
Pilot1	30.68	16.92	14.39	23.10	19.34
Pilot2	33.04	34.08	22.33	25.82	22.42
Pilot3-minus-LS	28.84	22.41	16.07	25.01	20.33
Δ +synset(node,sibling)					+0.10
Δ +synset&supersense(node,parent)					+0.11
Δ "fixed" entities (HideIT)	+0.79	+0.49	+1.00	+0.74	+0.37
Δ specialized lexicons (gazetteers)	+3.67	+3.47	+2.98	+3.02	+1.00
Δ adaptation by TM interpolation	+0.78	+6.75	+0.17	+0.92	+1.42
Δ terminology treelets	−0.04	+3.84	+2.50		+0.33
Δ VowpalWabbit	+2.35				
Δ total LS	+5.83	+11.77	+7.34	+4.60	+3.12
full Pilot3	34.67	34.18	23.41	29.61	23.45

Table 24: Translations from English (Batch2a). Effect of various lexical semantic modules on BLEU performance.

6 Final remarks

This deliverable has reported the experiments on further improving the quality of translations using Lexical Semantic techniques, including WSD and Linked Open Data (LOD) resources like WordNet or DBpedia[21] (the LOD version of Wikipedia). The information from online sources such as Wikipedia and a new transduction algorithm based on Vowpal Wabbit have been mildly successful. Regarding online sources, we have tried to deploy a sophisticated method to exploit comparable corpora in Wikipedia, which, although able to improve over baselines, provides similar improvement as that of a simpler technique, already incorporated in a previous iteration in Pilot 2, which uses just the names of Wikipedia pages and cross-Wikipedia links. Regarding the use of new transduction algorithms, we were not able to obtain positive results until the very last minute, and only for one language. We hope that in the future, those good results for Czech will carry over to other languages.

We have also reported improvements using a syntactically-annotated version of existing gazetteers, which shows that the use of deep techniques is able to improve results over shallow techniques in this setting.

Regarding the use of word sense information, the improvements discovered for English to Portuguese using TectoMT do not seem to carry over to other languages. On the contrary, further improvements of the WSD algorithm do produce improvement for English to Bulgarian when using a factored architecture. It seems that there is still room for improving the quality of MT using WSD techniques.

Some improvements due to deep techniques depend on features of the language pairs. This is the case of coreference, where the positive results when translating from English to Czech and Dutch do not carry over to Basque and Spanish, due to the different linguistic typology.

The improvements discovered in this final year are added on top of the successful techniques from previous years. As mentioned in D5.7, the results on the IT domain were improved treating domain-specific entities like URLs and interface commands. The incorporation of gazetteers mined from Wikipedia and DBpedia are highly beneficial, specially when using translation model interpolation. All in all, lexical semantics accounted for consistent improvements, with large improvements in many cases, specially for translations from English.

In addition, QTLeap has made a large effort in curating and producing linguistic processing tools for the six languages covered in WP5 (Basque, Bulgarian, Czech, English, Portuguese and Spanish), including PoS taggers, lemmatizers, Named-Entity Recognition and Classification, Word Sense Disambiguation, Named-Entity Disambiguation and Coreference software. The tools, their evaluation and the corpora annotated with those tools are fully described in Deliverable D5.9. In addition, D5.9 described the additional improvements to Word Sense Disambiguation (Bulgarian, Czech, English and Spanish) and Named Entity Recognition and Classification (Basque, English and Spanish). This deliverable now extends those improvements to Named Entity Disambiguation (English). All in all, QTleap has advanced the state-of-the-art of publicly available tools for those six languages.

[21] http://dbpedia.org

References

Eneko Agirre, Ander Barrena, and Aitor Soroa. Studying the wikipedia hyperlink graph for relatedness and disambiguation. *CoRR*, abs/1503.01655, 2015. URL http://arxiv.org/abs/1503.01655.

Ander Barrena, Eneko Agirre, Bernardo Cabaleiro, Anselmo Peñas, and Aitor Soroa. "one entity per discourse" and "one entity per collocation" improve named-entity disambiguation. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2260–2269, Dublin, Ireland, August 2014. Dublin City University and Association for Computational Linguistics. URL http://www.aclweb.org/anthology/C14-1213.

Ander Barrena, Aitor Soroa, and Eneko Agirre. Combining mention context and hyperlinks from wikipedia for named entity disambiguation. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*, pages 101–105, Denver, Colorado, June 2015. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/S15-1011.

Alberto Barrón-Cedeño, Cristina España-Bonet, Josu Boldoba, and Lluís Màrquez. A Factory of Comparable Corpora from Wikipedia. In *Proceedings of the 8th Workshop on Building and Using Comparable Corpora (BUCC)*, pages 3–13, Beijing, China, July 2015.

Bernd Bohnet. Very high accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics*, COLING '10, pages 89–97, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1873781.1873792.

Katrin Erk. A simple, similarity-based model for selectional preferences. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 216–223, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P07-1028.

Rosa Gaudio and Antonio Branco. Using wikipedia to collect a corpus for automatic definition extraction: comparing english and portuguese languages. In *Anais do XI Encontro de Linguistiva de Corpus - ELC 2012*, Instituto de Ciências Matemáticas e de Computação da USP, em São Carlos/SP, 2012.

X. Han and L. Sun. A generative entity-mention model for linking entities with knowledge base. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 945–954, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 978-1-932432-87-9. URL http://dl.acm.org/citation.cfm?id=2002472.2002592.

J. Hoffart, M.A. Yosef, I. Bordino, H. Fürstenau, M. Pinkal, M. Spaniol, B. Taneva, S. Thater, and G. Weikum. Robust Disambiguation of Named Entities in Text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 782–792, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 978-1-937284-11-4. URL http://dl.acm.org/citation.cfm?id=2145432.2145521.

John Langford, Lihong Li, and Alex Strehl. Vowpal wabbit online learning project, 2007.

David D Lewis, Yiming Yang, Tony G Rose, and Fan Li. Rcv1: A new benchmark collection for text categorization research. *The Journal of Machine Learning Research*, 5:361–397, 2004.

Rada Mihalcea and Dan I. Moldovan. extended wordnet: progress report. In *in Proceedings of NAACL Workshop on WordNet and Other Lexical Resources*, pages 95–100, 2001.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

George A. Miller, Claudia Leacock, Randee Tengi, and Ross T. Bunker. A semantic concordance. In *Proc. of HLT '93*, pages 303–308, 1993. ISBN 1-55860-324-7. doi: http://dx.doi.org/10.3115/1075671.1075742. URL http://dx.doi.org/10.3115/1075671.1075742.

Philip Resnik. Selectional constraints: An information-theoretic model and its computational realization. *Cognition*, 61:127–159, November 1996.

Felix Sasaki, Tatiana Gornostay, Milan Dojchinovski, Michele Osella, Erik Mannens, Giannis Stoitsis, Phil Ritchie, and Kevin Koidl. Introduction to FREME: Data meets language meets business. In *Proceedings of the EU Project Networking track at the 12th Extended Semantic Web Conference (ESWC2015)*, 2015.

Kiril Ivanov Simov, Petya Osenova, and Alexander Popov. Using context information for knowledge-based word sense disambiguation. In *Artificial Intelligence: Methodology, Systems, and Applications - 17th International Conference, AIMSA 2016, Varna, Bulgaria, September 7-10, 2016, Proceedings*, pages 130–139, 2016. doi: 10.1007/978-3-319-44748-3_13. URL http://dx.doi.org/10.1007/978-3-319-44748-3_13.