INFORMATICA, 2008, Vol. 19, No. 4, 535–554

Multi-Alignment Templates Induction

Algirdas LAUKAITIS, Olegas VASILECAS
Faculty of Fundamental Sciences, Vilnius Gediminas Technical University
Saulėtekio al. 11, LT-10223 Vilnius, Lithuania
e-mail: algirdas.laukaitis@fm.vgtu.lt, olegas.vasilecas@fm.vgtu.lt

Received: January 2008; accepted: June 2008

Abstract. This paper examins approaches for translation between English and morphology-rich languages. Experiment with English–Russian and English–Lithuanian revels that "pure" statistical approaches on 10 million word corpus gives unsatisfactory translation. Then, several Web-available linguistic resources are suggested for translation. Syntax parsers, bilingual and semantic dictionaries, bilingual parallel corpus and monolingualWeb-based corpus are integrated in one comprehensive statistical model. Multi-abstraction language representation is used for statistical induction of syntactic and semantic transformation rules called multi-alignment templates. The decodingmodel is described using the feature functions, a log-linear modeling approach and A^* search algorithm. An evaluation of this approach is performed on the English–Lithuanian language pair. Presented experimental results demonstrates that the multi-abstraction approach and hybridization of learning methods can improve quality of translation.

Keywords: machine translation, natural language processing, statistical induction, EM algorithm, A^* search.

1. Introduction

The development of the translation system between English and morphology-rich language pair can be a challenging task. Traditional approach for such development will be acquisition of the dictionary and transformation rules that are build manually by experienced linguist. The big advantage of such approach is a reliability of expert knowledge. The biggest shortcoming of such approach is an expensiveness of development and maintenance of manually build rules base (see VDTTranslate for an evaluation of English–Lithuaniant translation based on manually build rules base (English–Lithuanian translation, 2008)).

Corpus based machine translation is an alternative approach. There are two main directions in corpus based machine translation: statistical machine translation (SMT) and example based machine translation(EBMT) (see (Somers, 1999) for EBMT review and (Och and Ney, 2002) for the state-of-art SMT). The idea for EBMT dates from the paper (Nagao, 1984) and consist of the three main components: matching phrases against a database of examples, identifying the corresponding translation phrases, and then recombining these to give the target text. The problem with the EBMT when a new language pair is considered is that the longer the matched phrases are, the lower the probability of

a match will be. The millions of matched phrases are required to build reasonable translation system. Our experiment with Russian and Lithuanian languages reveled that even by taking two words phrases (bigrams) we need more that 10 million phrases to cover 75% of new randomly chosen text from Internet. To build such amount of bilingual phrases by hand or by half automated process is difficult and for the most languages unachievable goal.

Another corpus based approach is the statistical machine translation. This approach is dated from the works presented in the paper (Brown *et al*., 1993). Differently from EBMT, examples are not stored at all in SMT systems. Instead of examples, the precomputed statistical parameters which give the probabilities for bilingual word pairings are stored. However, the trend towards statistical phrase and generalized phrase started to dominate in the statistical paradigm and many ideas presented by the EBMT researches are reinvented by the statistical means. But, for morphology-rich language such as Lithuanian with more than 100 million different legitimate word forms generated from 100 thousand lemmatized words from dictionary, straightforward inference based on SMT approach is little of use (Section 2 demonstrates that). Given all remarks presented above we can formulate the main goal of this paper:

Search for all reliable linguistic resources that can be attainable for free on the Web. Development of the statistical machine translation learning and decoding framework with all available linguistic knowledge.

Fig. 1 explains the general idea that we formulated to solve this task. The classical translation pyramid is presented in the left side of the Fig. 1. Fig. 1 right side explains the main idea suggested in this paper. Direct translation, syntax, and semantic levels from classical pyramid are split into many hierarchically ordered levels. Then, the statistical machine learning methods are used for search of transformation templates on each of those levels. Framework presented in this paper can be viewed as a fusion of many ideas established in SMT and EBMT areas.

The rest of the paper describes suggested approach in details and is organized as follows. In Section 2 we present experiment on English–Russian and English–Lithuanian

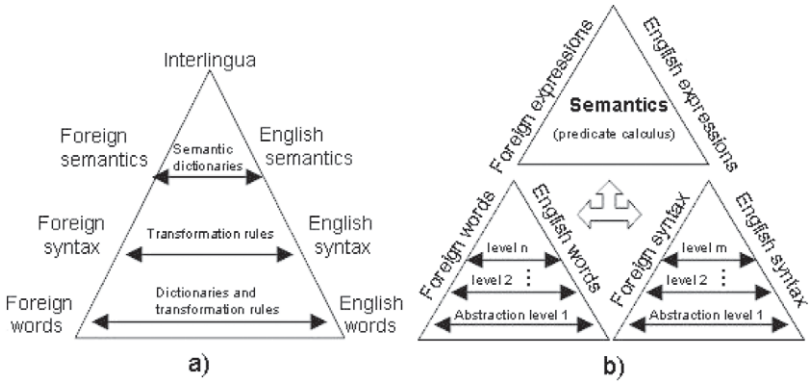

Fig. 1. Splitting the classical translation pyramid.

translation when "pure" statistical methods are used and the only resource of learning data is a parallel corpus of approximately 1 million words for Russian and 10 million words for Lithuanian languages. Further explanations are given why any of existing methodologies are failing to build reliable translation system. The state-of-art languages processing resources and multi-lexical-syntactic language representation is suggested in the Section 3. Section 4 presents a statistical framework of the multi-alignment temples that can be induced from bilingual corpus. Evaluation results are given in the Section 6. Concluding remarks and future work are presented at the end of the paper.

2. Motivation for MT Hybrid Approach: Case Study

The current interest in the machine translation world is the rapid development of systems for novel language pairs when often on of the language in pair is English. There was promising results reported by the Brown group 15 ago (Brown *et al*., 1993) on the automated translation development using statistical inference methods. Yet, the experiments with 11 European Languages (Koehn, 2005) demonstrated that Finish language which is morphologically most complex among tested gives the worst translation results. There were more positive results for other tested languages but all of them belongs to the Germanic and Italic group.

Until recently there was little research done for the Slavic and Baltic group of languages (Vaiciunas and Raskinis, 2006). Report in (Laukaitis and Vasilecas, 2007a) on English–Lithuanian language pair confirmed that traditional automatic methods gives unsatisfactory results when the corpus is the only resource for learning. In this section we report on test done on English–Russian and English–Lithuanian language pairs. We used the IBM Model 3 for statistical inference (Brown *et al*., 1993) and the breadth-first search algorithm for decoding.

Russian language has received much of attention in the domain of machine translation and there is number of translation systems for Russian language. But, there is only a small parallel corpus of English–Russian languages available on the Internet (Russian–English parallel texts, 2002). We used this corpus to build our own English–Russian $\sim$1 million words corpus for each language pair. The BLEU score (Kishore *et al*., 2002) that we received after training the system was $\sim$0.02 which shows that the system is not capable to present any meaningful translation.

We understand that the Russian–English corpus was too small for suitable machine translation. Then,we were able to get better results for English–Lithuanian language pair because the corpus we compiled was 10 times bigger than English–Russian corpus. Test results for this language pair are presented in Fig. 2. We used the EuroLex (Official Journal of the European Union, 2006) corpus ($\sim$9.8 million words for Lithuanian and $\sim$12.1 million words for English) for training IBM Model 3. Then we tested it on three different texts with 1000 sentences each with 18 words on average. We can see that testing set from EuroLex corpus (marked *Euro* in Fig. 2) scored higher than testing set from news bulletin (marked *News* in Fig. 2) and much higher than testing set from "Robinson Crusoe" a

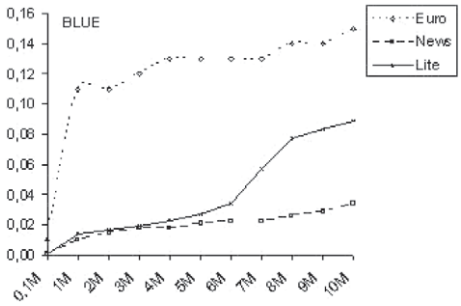

Fig. 2. Three testing sets case. Translation quality dependency from training corpus size.

novel by Daniel Defoe (marked *Lite* in Fig. 2). Fig. 2 suggest one important conclusion: if the training is done on the corpus from only one domain then the testing results from different domains can differ significantly.

Our explanation for such bad translation results is that the learned lexicon and parameters form ∼1 million (Russian language case) or ∼10 million (Lithuanian language case) words from one domain corpus is insufficient for general purpose translation system. Nevertheless, test results reveled that "pure" statistical approach between different groups of languages can be unfeasible task with the current parallel corpus on the shelf (exception can be English–Chines language pair because there is ∼150 million words corpus available). For the explanation of received result we suggest to consider following facts about English–Lithuanian language pair. They can explain why translations were so bad.

1. Given a dictionary of ∼100 000 lemmatized words (lexemes) the languages like English or French will generate up to ∼400 000 unique words (word-forms). The limit for such morphology complex language like Lithuanian will generate more than ∼100 000 000 unique words (word-forms) from ∼100 000 lemmatized words (lexemes). Then statistical learning of the translation probability parameters is faced with the fact that most of the words remains out of learning scope.
2. Moderate size parallel corpus like one used in this project (∼9.8 million words for Lithuanian and ∼12.1 million words for English) gives ∼43 000 unique words (word-forms) received from ∼22 000 English language lexemes in this corpus. Respectively, Lithuanian language gives ∼220 000 distinct words received from ∼21 000 lexemes. Then, much of linguistic information remains unlearned.

3. Natural Language Processing Resources and Multi Level Representation

Natural language is an example of sparse and high-dimensional representation in the domain of machine learning. The spareness of data can be handled by using additional information and especially one where hierarchical relationships can be established. The challenge for the translation task is that the similar resources must be established on both sides of translation, i.e., for each translation language. Fig. 3 shows the final structure of

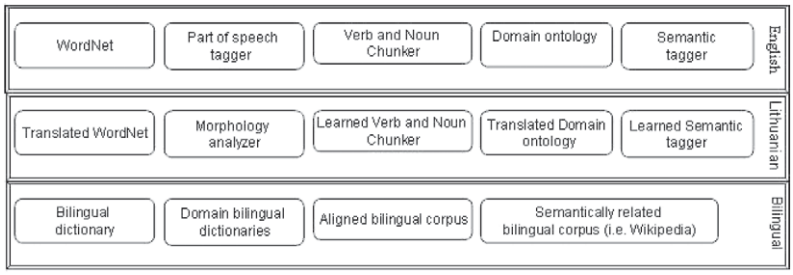

Fig. 3. Identified linguistic resources.

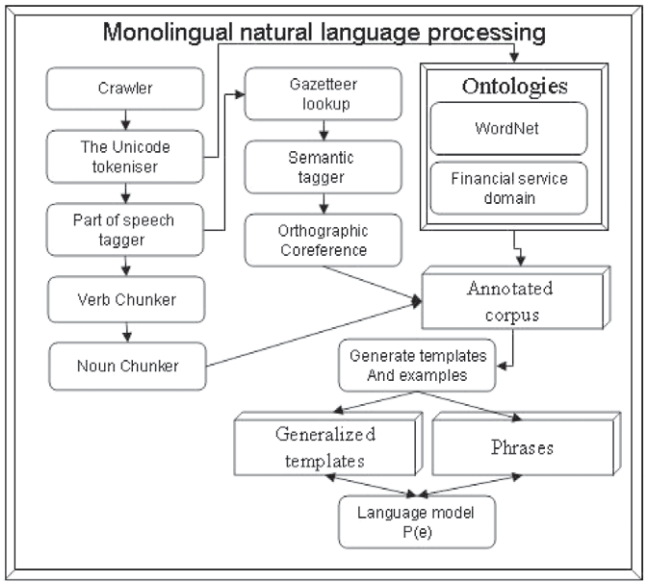

Fig. 4. Natural language procesing.

the resources we identified in the preliminary stage of this research project. The resources are divided into three categories: English language resources, Foreign language resources (i.e., Lithuanian), Bilingual translation resources. Anyone familiar with natural language processing (NLP) will see that for English language those resources can be easy acquired for free on the Web (all NLP processes are shown in Fig. 4). For the rest of the languages, even for the major languages like Russian, it is diff icult to obtain those resources (we investigated Russian language as one of the world major language and we where surprised that neither POS tagger or semantic dictionary are unobtainable from the Web). Then, in this section we discuss the English language resources and we will give some remarks how to build similar resources in one man-month for rest of languages using OpenOffice spell checking data and Internet as monolingual corpus.

WordNet (Miller, 1985) is the words ontology base that contains words semantic rela-

tionships in the synset form, a set of synonyms representing a distinct concept. Additionally it describes words hypernyms-hyponyms-synonyms relationships that can help us to measure words semantic distances. Those distances has an important role for the machine learning algorithm presented in the next section. Additionally, using the WordNet ontology we extend the size of the convention ∼100.000 unique words bilingual dictionary with ∼200.000 translation records up to several million translation records where each translation has the same semantic distance metric as it is described by the WordNet dictionary (for English–Lithuanian language the extended translation dictionary concept can be tested at (Lithuanian language translation, 2007)).

The second important issue of ontology uses for translation is related to the abstracted language representation. The rule-of-thumb that we suggest in the current paper is described as follows:

1) *take a corpus*,
2) *chose the most frequent noun from this corpus*,
3) *look at WordNet for this particular word and collect all words that are included in the set of senses of this particular word as well as all the words that are included in set of hyponyms of this particular word*,
4) *mark all those collected words as one class*,
5) *transform all marked words from the given corpus by replacing them with the label of received class and then iterate from the beginning until all nouns are mapped.*

We can receive numerously number of such representations by defining various distance metrics between the words in the WordNet dictionary. Table 1 shows the examples of abstracted representation using WordNet ontology. Additionally to the WordNet ontology Domain specific otologies and semantic taggers can be used. The techniques for utilizing them are the same as with WordNet ontology described above. All those notions with WordNet, Domain specific ontologies or semantic taggers can be attributed to the bottom-left pyramid in Fig. 1 where translation is done between English and Foreign words on several abstraction levels. More examples on multi-hierarchical representation for other areas of natural language processing can be found in (Laukaitis and Vasilecas, 2007b) or (Laukaitis and Vasilecas, 2007c).

Now, we can ask the question: How to receive the same cardinality of abstracted representation on the Foreign language side. In this paper we suggest the concept of *Translated WordNet* to answer this question.

Table 1

Sentence incremental abstraction using WordNet ontology

Level	Representation
0	Later, in the Bohemian crowd of San Francisco, he was called Kit Bellew.
1	Later, in the Bohemian [gathering] of San Francisco ...
2	Later, in the Bohemian [social group] of San Francisco ...
...	
i	Later, in the [regional] [group] of [Location], he was [identified] [Person].

Translated WordNet. There is two steps in developing WordNet analogy on the Foreign language side. The first step represents a naive translation of English language WordNet. We assume that there is bilingual dictionary that covers all English language words included in WordNet. The second step of the *Translated WordNet* development is manual correction when errors and disambiguations are removed. The second step is continuous work on quality improvements. When significant improvements are received new translation training cycle can be launched.

Next, the following rule can be used to receive an abstracted corpus with close granularity as one received on the English language side:

1) *The OpenOffice spell checking resources are used to map all morphology rich language words to their basic lemmatized forms.*

2) *Then, by choosing previously mapped English language classes and Translated WordNet dictionary we map all foreign words to their English language counterparts.*

Part of Speech tagger. Many Part of Speech (POS) taggers have been written for the English language. Nevertheless, when someone is building Web based information system that utilises linguistic resources, additional requirements are imposed on the POS tagger components. One of the most important requirement is suitability to integrate linguistic components with the other components of information system. We found that one of the best currently available solution which meets the requirements for linguistic analysis quality and requirements for information system integration is the GATE NLP system (Cunningham, 2002). Additionally to the POS tagger we used tagger of noun and verb phrases to generate several lexical-syntactic language representations.

There is number of ways to abstract language using POS, noun and verb phrases taggers. Table 2 presents several examples of the lexical-syntactic abstractions. We can see that at the level 1 all open class words (nouns, verbs, adverbs and adjectives) are replaced respectively by their POS tags labels. At the level 2 all noun phrases are replaced by "NC" label. At the level 3 all close class words are replaced respectively by their POS tags labels.

We have several lexical-syntactic English language representations. Then the question is: How useful are those representations for the translation between English and

Table 2

Lexical-syntactic incremental abstraction of sentence from "Robinson Crusoe" a novel by Daniel Defoe using GATE NLP system

Level	Representation
0	I found that the forty bushels of barley and rice were much more than I could consume in a year
1	I SimPasFVGactive that the NN NNS of NN and NN SimPasFVGactive RB JJR than I noneMODALnone in a NN
2	NC SimPasFVGactive that the NC of NC SimPasFVGactive NC than NC noneMODALnone in NC
3	NC SimPasFVGactive IN NC IN NC SimPasFVGactive NC IN NC noneMODALnone IN NC

morphology rich language? Another question: How to build adequate representations in the Foreign language, not just in English?

The answer for the first question refer to our ability to induce transformation rules that uniquely maps from abstracted English language to abstracted Foreign language and vice versa. In other words we can say that English and Foreign language abstract representation must be isomorphic. Isomorphism in this case is understood as the ability to induce corresponding mapping between linguistic structures in such a way that there always will be an example on the natural language level which can be treated as correct translation. If we use such translation isomorphism definition then, we can say that abstraction level 3 in the Table 2 is not an isomorphic representation because such representation loses too much information by POS-tagging all close-form words. Close-form words in English language plays an important role in determining what word morphological class will be used when translating between English and morphology rich language and then they must be left unchanged for the transformation templates.

The next question is how to build adequate lexical-syntactic representations for the Foreign morphology rich language. At first, we need morphology analyser which grammatically classifies words according grammar textbooks. Ten, we need a mapping that describes Foreign grammar classes in terms of English language grammar. After that, representation is straightforward as one for the English language above.

It is not in the scope of this paper to describe how to build such analysers, but there are two hints that helps to build such analyser in short period time:

- Acquire OpenOffice spell checking files and then generate all possible legitime word forms for that particular language (for Lithuanian we received more than 100 million different words).
- Use OpenOffice spell checking rule file and short description of language grammar to build final set of rules.

Next section describe how to induce translation templates from those multi-level language representations once you have all resources described in this section.

4. Multi-Alignment Templates Induction

The statistical approach for the machine translation is an attractive method because it suggest fully automated process of learning. Nevertheless, in Section 2 we demonstrated that "*pure*" statistical methods are insufficient for practical translation between English and morphology rich languages. Then, hybridization by incorporating linguistic resources into statistical framework can be the answer. In this paper the idea of factoring probability distribution of one language given another is extended by incorporating all available linguistic information. The main extension is probability factoring on multi-level language representation. Let e_i stands for English and f_j for foreign language representation on language abstraction levels i and j, where $i, j \in [0 \dots N]$ and N is a number of language representation levels. Then the starting point for probability factoring will be

$$P(f_j)P\big(e_i|f_j\big) = P(e_i)P\big(f_j|e_i\big). \tag{1}$$

The decision on e_i given f_j can be made without regarding $P(f_j)$

$$\hat{e}_i^j = \arg\max_{e_i} P\big(e_i|f_j\big) = \arg\max_{e_i} P(e_i)P\big(f_j|e_i\big). \tag{2}$$

Another way in which we can deside upon e_i is to consider all linguistic information in possession which we model as language representation on many abstraction levels

$$\hat{e}_i = \arg\max_{e_i} P\big(e_i|f_0,\ldots,f_N\big) = \arg\max_{e_i} P(e_i)P\big(f_0,\ldots,f_N|e_i\big). \tag{3}$$

Then, one way to follow will be the development of parametric form for the $P(e_i|f_i)$ and $P(e_i|f_0,\ldots,f_N)$. In many problems that adopts Bayesian framework of modeling the hidden parameters are introduced to reduce the dimensionality of the model. For the statistical machine translation alignments are used as a hidden parameter. The first time alignments where introduced in IBM models 1–5 (Brown *et al*., 1993) where expectation maximization (EM) algorithm was used to estimate model parameters. The only required resource for IBM models 1–5 was bilingual, sentence aligned corpus. The French–English language pair has demonstrated that by using those models it is possible to obtain satisfiable translation. Nevertheless, it was demonstrated in Section 2 that English and morphology rich language is a more difficult case.

In this paper we assume that we have the linguistic resources described above. Those linguistic resources lets us simplify alignments interpretation compered with it's original introduction in (Brown *et al*., 1993), i.e., alignments are assumed to be one-to-one mapping between lexical units. Mapping between lexical units is understood as mapping between word to word, word to phrase or prase to phrase. It helps us avoid such '*strange*' parameters as fertility of the word or spontaneous appearance of the word. It is appropriate to mention that one-to-one mapping between lexical units is close to symmetrized alignments presented in (Och and Ney, 2002).

With alignments a probability (1) can be factored as follows:

$$P(f_i)P\big(e_i|f_i\big) = P(e_i)\sum_a P\big(f_i,a|e_i\big), \tag{4}$$

or if all linguistic information is considered

$$P(f_0,\ldots,f_N)P\big(e_i|f_0,\ldots,f_N\big) = P(e_i)\sum_a P\big(f_0,\ldots,f_N,a|e_i\big). \tag{5}$$

At this point it is important to make the following note. In all equations from 1 to 5 we used subscript i to mark abstraction level on which translated string e_i is received. Usually translation is assumed on the natural language level which we mark as $i = 0$, i.e., e_0. Nevertheless, by using WordNet we can receive many abstracted natural language representations and all probability factoring used up to this point will be correct.

4.1. *Lexical Alignments*

The English language string '*he is living in the house*' can be translated in the Lithuanian language like '*jis gyvena name*'. In Fig. 5 a) we can see the traditional alignments as it was presented in (Brown *et al*., 1993) as IBM Model 3. Alignments assigns a single Lithuanian string position to some English word. If two English words align to the same Lithuanian word, then that Lithuanian word is said to have a fertility of two. If a word remains unaligned, then it has fertility zero. There is an invisible word *null* which produces spurious words in the target language.

In Fig. 5 part b) we explain the one-to-one alignments approach. At the natural language level (which we call abstraction level 0) we align all words that are in the bilingual dictionaries. If there remains unaligned words then they are the candidates to become part of the phrase where the phrases are aligned in one-to-one fashion. Received one-to-one alignments are similar to what authors in (Och and Ney, 2002) called symmetrized alignments. Next, we describe one-to-one alignments algorithm in details.

Let $e = e_1 e_2 \dots e_l$ be English language sentence and $f = f_1 f_2 \dots f_m$ a foreign language sentence. Let D be a bilingual dictionary consisting of the pairs of words $(\dot{e}, \dot{f})$, where $\dot{e}$ is a word in English and $\dot{f}$ is a word in foreign language. Let $dictranslate(\cdot)$ be a function that returns the set of the word translations from dictionary D, $lemmas(\cdot)$ be a function that returns set of lemmas of the word and $semantics(\cdot)$ be a function that returns set of hypernyms, hyponyms and synonyms of the word. Additionally, functions $semantics(\cdot)$ and $lemmas(\cdot)$ returns empty word (null) value if a given word is from close class of words (i.e., if the word is determiner, conjunction, particle etc.). We define the function $match(wordset1, wordset2)$ which returns true if there is at least one word in the set $wordset1$ and in the set $wordset2$.

Then, with sentences e and f from parallel corpora we consider the set of alignments $A = \{i, j\},\ i \in [1, .., l],\ j \in [1, .., m]$ where one of the followings holds:

1. $match(dictranslate(e_i), f_j) = true$.

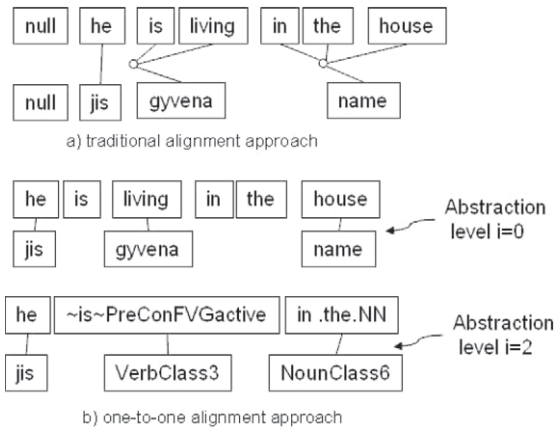

Fig. 5. Alignments: traditional and one-to-one approaches.

2. $match(dictranslate(lemmas(e_i)), f_j) = true.$
3. $match(dictranslate(e_i), lemmas(f_j)) = true.$
4. $match(dictranslate(lemmas(e_i)), lemmas(f_j)) = true.$
5. $match(dictranslate(semantics(e_i)), lemmas(f_j)) = true.$

The alignments $A = \{i, j\}$ defined by those constrains uses only conventional dictionary and semantic dictionary, i.e., WordNet. Then 5 can be written as

$$P(f_0)P\big(e_0|f_0\big) = P(e_0)\sum_a P\big(f_0, a|e_0\big). \tag{6}$$

The rest of linguistic information (i.e., noun and verb chunking and POS tagging) is put in the form of the following constraints:

1. We require that English language noun phrase must be translated in compact word sequence of the foreign language.
2. The same must hold for English language verb phrase.
3. Manually build rules are used to say which English language POS tag is matched with which foreign language POS tag.

Then, we mark $f = f_0$ and $e = e_0$ so that (6) is written in conventional way

$$P(f)P\big(e|f\big) = P(e)P\big(f|e\big) = P(e)\sum_a P\big(f, a|e\big). \tag{7}$$

$P(f, a|e)$ can be decomposed in many ways. Let d_i be the distance between English word e_i and it's aligned translation f_{a_j}. Then, probability $P(f, a|e)$ can be decomposed similarly as IBM Model 1:

$$P\big(f, a|e\big) = \epsilon\prod_j^m \frac{1}{d_{a_j}} t\big(f_j|e_{a_j}\big). \tag{8}$$

By defining indicator function $I(a_j)$ which is equals 1 when match is found and 0 otherwise we write

$$P(f|e) = \epsilon\sum_{a_1}^{l}\cdots\sum_{a_m}^{l}\prod_j^m \frac{1}{d_{a_j}} t\big(f_j|e_{a_j}\big)I(a_j), \tag{9}$$

subject to

$$\sum_f t(f|e) = 1. \tag{10}$$

Unconstrained auxiliary function

$$h(t|\lambda) = \epsilon\sum_{a_1}^{l}\cdots\sum_{a_m}^{l}\prod_j^m \frac{1}{d_{a_j}} t\big(f_j|e_{a_j}\big)I(a_j) - \sum_e \lambda_e\Big(\sum_f t(f|e) - 1\Big). \tag{11}$$

The partial derivative of h with respect to $t(f|e)$

$$\frac{\partial h}{\partial t(f|e)} = \epsilon \sum_{a_1}^{l} \dots \sum_{a_m}^{l} \sum_{j}^{m} \delta(f, f_j)\delta(e, e_{a_j}) t(f|e)^{-1} \times \prod_{k}^{m} \frac{1}{d_{a_k}} t\big(f_k|e_{a_k}\big) I(a_k) - \lambda_e. \tag{12}$$

If this partial derivative is zero then

$$t(f|e) = \lambda_e^{-1} \epsilon \sum_{a_1}^{l} \dots \sum_{a_m}^{l} \sum_{j}^{m} \delta(f, f_j)\delta(e, e_{a_j}) \prod_{k}^{m} \frac{1}{d_{a_k}} t\big(f_k|e_{a_k}\big) I(a_k). \tag{13}$$

Alignments A that maximizes $P(f, a|e)$ are called Viterbi alignments. Let mark them as in (Brown *et al*., 1993) $V(f|e)$. An iterative estimation of Viterbi alignments by using (13) we mark as $\hat{V}(f|e)$. At this point it is important to remaind that neither f or e are treated as translations of each other. They are rather an anchor points that gives us alignments on open class words found in conventional and WordNet dictionaries. Next, lexical-syntactic alignments are introduced. Those alignments segments English and foreign language sentences into phrases and then one-to-one mapping is received on the bases of those phrases.

4.2. *Lexical-Syntactic Alignments*

Let $\hat{V}(f|e)$ be an estimated open class words alignments as defined above. Let $W_e^{(s)}$ be the set of unaligned English words in the sentence s and $W_f^{(s)}$ be the set of unaligned foreign words in the sentence s. The following assumption about $\hat{V}(f|e)$ defines steps of algorithm.

We assume that $\hat{V}(f|e)$ is such that the rest of unaligned words can be aligned by attaching them either to the left or right of previously aligned words. If attachment to the left we mark as -1 and to the right as 1 then we receive a binary alignments to the previously aligned anchor words. Such binary alignments will be random variables which will be estimated as previously with EM algorithm. The Eqs. 6–13 are used in the same way if the following interpretations are introduced:

1. f and e means abstracted representation sentences. E.g., from Fig. 5 part b) on the natural language level we have $e =$ ‘he is living in the house‘ and $f =$ ‘jis gyvena name‘. Then, interpretation of e and f on lexical-syntactic level will be $e =$ ‘he is PreConFVGactive in .the.NN‘ and $f =$ ‘jis VerbClass3 NounClass6‘.
2. a means a binary alignments.
3. $I(\cdot)$ is interpreted as an indicator that controls where word points at nearest anchor word.

4.3. *Lexical-Syntactic Feature Functions*

The Eq. (5) is only one of many ways we can start to factorize posterior probability $P(e_i|f_0,\ldots,f_N)$. For the decoding process, in this paper we use direct posterior probability modeling approach via the subset of feature functions. The framework was suggested for natural language processing by (Berger *et al.*, 1996) and there we reformulate it for specific language representation on abstraction level i:

$$P\big(e_i|f_0,\ldots,f_N\big) = \frac{1}{Z_\lambda(f)} \exp\left(\sum_{j=1}^{M} \lambda_m h_m(e_0,\ldots,e_N,f_0,\ldots,f_N)\right), \tag{14}$$

where $h_m(e_0,\ldots,e_N,f_0,\ldots,f_N)$, $m = 1\ldots M$ are the feature functions and λ_m are model parameters that are found from parallel corpus training. Function Z is a normalization constant determined by the requirement $\sum_{e_i} P(e_i|f_0,\ldots,f_N) = 1$ and is equal

$$Z_\lambda(f) = \sum_{e_i} \exp\left(\sum_{j=1}^{M} \lambda_m h_m(e_0,\ldots,e_N,f_0,\ldots,f_N)\right). \tag{15}$$

The expression (14) in some way is close to an heuristic modeling approach used by EBMT approach, i.e., feature function in (14) can be build on the EBMT heuristics. Again, when translating $f_0,\ldots,f_N$ (i.e., f_0 stands for natural language representation and $f_1,\ldots,f_N$ stand for abstracted representatons) we chose the translation $\hat{e}_i$ which is the most probable under learned model:

$$\hat{e}_i = \arg\max_{e_i} \sum_{j=1}^{M} \lambda_m h_m(e_0,\ldots,e_N,f_0,\ldots,f_N). \tag{16}$$

Parameters λ_m are found using maximum-entropy principle:

$$\hat{\lambda}_m = \arg\max_{\lambda_m} \sum_{s=1}^{S} \log P\big(e_i|f_0,\ldots,f_N\big), \tag{17}$$

where S is a number of sentences in bilingual corpus used for the training.

If alignments are used then $P(f_0,\ldots,f_N,a|e_i)$ can be modeled using direct posterior probability framework as well:

$$P\big(f_0,\ldots,f_N,a|e_i\big) = \frac{1}{Z_{\lambda,a}(f)} \exp\left(\sum_{j=1}^{M} \lambda_m h_m\big(e_0,\ldots,e_N,f_0,\ldots,f_N,a\big)\right). \tag{18}$$

For the evaluation of λ_m parameters we used the iterative scaling algorithm. Basic steps of this algorithms for the problem at hand can be found in (Berger *et al.*, 1996).

Translation Model Feature Functions

In the Section 3 we described resources that we would like to use for posterior probability modeling. For that purpose we employ feature functions as a flexible way to code-in that knowledge. For the representation of the natural language on several lexical-syntactic levels we reinterpret alignment templates idea given in (Och and Ney, 2002) and we induce alignment templates on several levels of language representation. Formally alignment template z is a triple $(F_1^J, E_1^L, \tilde{A})$ that describes the alignment $\tilde{A}$ between source class sequence F_1^J and target class sequence E_1^L (see (Och and Ney, 2002)). Our reinterpretation consists from the following two steps:

1. Instead of one abstraction level we introduce several abstraction levels and then formally z_i means alignment template on abstraction level i.
2. Syntactic and lexical information is used to determine word classes. In this research four levels of representation are defined: 0) Natural language level. No replacements are done ($i = 0$). 1) WordNet and translated WordNet based abstraction ($i = 1$). 2) Level where all nouns, verbs, adjectives, adverbs are labeled by their POS labels and the rest of words (close-form words) are left unreplaced ($i = 2$). 3) All noun and verb phrases are identified and labeled using GATE system notations. Close-form words are left unreplaced ($i = 3$).

Then the language model feature functions are defined by multiplying the probability of all used alignment templates on each level separately and taking the logarithm:

$$h_{AT}^i(e^i, f^i, z^i) = \log \prod_{k=1}^{K} p\big(z_k^i | f^i\big). \tag{19}$$

Language Model Feature Functions

Typically for the language model the word-based n-gram models are used. In this paper language model feature functions are defined on each level of abstraction. By defining the level i and the phrase length n_i (i.e., n_i-gram in level i) we define the following language model feature functions:

$$h_{LM}^i = \log \prod_{l}^{L} p\big(e_l^i | e_{l-1}^i, \ldots, e_{l-n_i+1}^i\big), \tag{20}$$

where L is the length of the sentence and $p(e_l^i | e_{l-1}^i, \ldots, e_{l-n_i+1}^i)$ is the conditional probability of the word/phrase e_l^i on the abstraction level i. The rule of thumb is that more abstract representation we have then bigger n_i can be defined. In this paper for $i = 0$ and $i = 1$ a trigram language model has been used, for $i = 2$ a 4-gram and for $i = 3$ – a 5-gram class-based language model has been used.

5. Decoding

A decoding algorithm is important to the success of translation system. The role of the decoder is to find the translation that is most likely according

$$\hat{e} = \arg\max_{e} P(e|f) = \arg\max_{e} P(e)P(f|e). \tag{21}$$

For simple model like IBM Model 1 the decoding problem is NP-complete (Knight, 1999). Various heuristics and ad hoc solutions lets reduce search in polynomial time. In this paper, we consider decoder which incorporates all linguistic knowledge that we used in the training process. We use the A* search algorithm which is one of best-first search algorithms. The A* algorithm builds hypothesis incrementally and stores them in the ordered stack. The basic steps of the algorithms for the machine translation are as follows:

1. Initialize an empty ordered stack (highest score hypothesis it put at the top of the stack).
2. Take the best hypothesis from the top off the stack.
3. If it is a complete sentence then terminate and return translated sentence.
4. For each possible next linguistic unit LU (i.e., word, phrase, translation template), extend hypothesis by adding LU and push the resulting hypotheses onto the stack.
5. Return to Step 2.

As pointed in (Germann *et al*., 2001), one difference between the decoding process in such applications like speech recognition and machine translation is that speech is always produced in the same order as its transcription. In machine translation the left-to-right relation rarely holds even for language pairs as similar as French and English. Nevertheless, in most cases it is possible to find admissible translation in witch permutations are local, i.e., we can consider the k_{max} the length of phrases where permutation of words is allowable and keep monotonous translation between those phrases. The parameter k_{max} is defined by the size of the training corpus and the linguistic properties of source and target languages. In our experiment we found that $k_{max} = 5$ is enough to receive acceptable translation quality.

We already mentioned that important property for A* search algorithms is an heuristic function. If we use n as position in the sentence translated up to the current moment then heuristic function is writhen as

$$f(n) = g(n) + h(n), \tag{22}$$

where g is an evaluation of the translation up to position n and h is expected cost of remaining translation. Additionally, A* algorithm requires that, $h(n) < h_a(n)$ where $h_a(n)$ is actual translation cost of remaining part of the sentence. $g(n)$ we defined as

$$g(n) = \left(1 + \sum_{j=1}^{M} \lambda_m h_m\big(e_0, \ldots, e_N, f_0, \ldots, f_N\big)\right)^{-1}, \tag{23}$$

where where h_m and λ_m the same as in (14). Function $h(n)$ can be constructed from the subset of functions $\{h_m\}$ and defined the as (23). If we set $h(n) = 0$ we have breadth-first search algorithm which guarantees optimal solution but which is the most expensive computationally. We found that computationally most efficiently is to use $h(n)$ defined as

$$h(n) = \left(1 + Mh_M(e_0, \ldots, e_N, f_0, \ldots, f_N)\right)^{-1}. \tag{24}$$

6. Evaluation Results

Training and testing data where chosen the same as in the motivation example presented in the Section 2. The main question we would like to answer with this evaluation is what impact various linguistic resources have on the quality of generated translation. The same as in the motivation study we use BLEU score (Kishore *et al*., 2002). BLEU measures the precision of unigrams, bigrams, trigrams and 4-grams with respect to reference translations. Additionally, we introduce a new metric in this evaluation study. We call it WordNet-BLEU and it resolves one drawback of original BLEU related to rejection of reasonable translations.

Mathematical expression of the WordNet-BLEU remains the same as ordinal BLEU and difference is in treatment what is a good match of n-grams. BLEU scores translation n-gram as a match only if it is exact match of referenced n-gram. Translated phrase 'nice house' will be marked as incorrect if referenced translation is something like 'stunning chalet'. Nevertheless, semantically both phrases are correct and in some degree have the same meaning. We can match those phrases by defining the sets of synonyms-hypernyms-hyponyms for each word that has been generated by translation engine and for each word in referenced translation. Then, the whole idea of the WordNet-BLEU score is to make match not between words but between those sets of the word semantic relationships.

Formally, the match by WordNet-BLEU metric we define as follows:

Referenced word $\check{f}$ generates the set of words arranged by synonyms-hypernyms-hyponyms relationships $\check{F}$ (see (Miller, 1985) for details). Translated word $\ddot{f}$ generates the set of words arranged by synonyms-hypernyms-hyponyms relationships $\ddot{F}$. The referenced phrase $\check{f}_0, \ldots, \check{f}_k$ matches translated phrase $\ddot{f}_0, \ldots, \ddot{f}_k$ by means of WordNet-BLEU metric if:

1. The set $\check{F}_m$ of each word $\check{f}_m$, $m \in 1 \ldots K$ has at least one word that is found in the set $\ddot{F}_m$. The match is of degree d if there is d consequent synonyms-hypernyms-hyponyms layers in the set $\ddot{F}$ matched by d consequent synonyms-hypernyms-hyponyms layers in the set $\check{F}$.
2. Both phrases $\check{f}_0, \ldots, \check{f}_k$ and $\ddot{f}_0, \ldots, \ddot{f}_k$ matches morphologically, i.e., each word $\check{f_m}$ morphological class equals to the word $\ddot{f_m}$ morphological class.

Now, returning to our question about linguistic resources impact on the translation quality we present Tables 3 and 4. The first row of the tables displays the BLEU score of test test translation when only bilingual corpus was used for translation engine training. The second row shows the BLEU score when bilingual corpus and conventional dictionary was used. Those BLEU scores confirms what we emphasized in the motivation

Table 3

English–Lithuanian and Lithuanian–English translation dependencies from used linguistic resources and corpus size (millions of words) (BLEU metric) (1 – only bilingual corpus used, 2 – bilingual corpus and conventional dictionary used, 3 – bilingual corpus, conventional dictionary and POS taggers used, 4 – bilingual corpus, conventional dictionary,POS taggers used and noun and verb phrase taggers used, 5 – bilingual corpus, conventional dictionary,POS taggers used, noun and verb phrase taggers and WordNet used)

Abst.\ Size	1	4	7	10	1	4	7	10
EN–LT					LT–EN			
1	0.11	0.12	0.12	0.13	0.11	0.13	0.13	0.14
2	0.14	0.14	0.15	0.16	0.16	0.16	0.18	0.19
3	0.25	0.25	0.26	0.26	0.26	0.27	0.31	0.29
4	0.28	0.29	0.29	0.31	0.34	0.35	0.36	0.36
5	0.28	0.31	0.32	0.33	0.35	0.37	0.37	0.39

Table 4

English–Lithuanian and Lithuanian–English translation dependencies from used linguistic resources and corpus size (millions of words) (WordNet-BLEU metric)

Abst.\ Size	1	4	7	10	1	4	7	10
EN–LT					LT–EN			
1	0.13	0.14	0.15	0.16	0.14	0.15	0.16	0.16
2	0.18	0.18	0.11	0.11	0.19	0.19	0.13	0.16
3	0.29	0.31	0.33	0.36	0.33	0.36	0.39	0.43
4	0.33	0.36	0.38	0.42	0.39	0.43	0.47	0.50
5	0.34	0.39	0.43	0.47	0.43	0.46	0.50	0.51

example: when translating between English and morphology rich language bilingual corpus of 10 million words is too small for a reasonable translation. The third row shows scoring when we additionally added POS taggers, i.e., morphological when knowledge base was added. We can see that scoring quality almost doubled. The main reason for this improvement is that we were able map all morphological forms to lemma forms. The forth row in the table shows scoring when noun and verb taggers were added. The las row in the ta bles shows scoring when WordNet dictionary has been added. We can see that phrases taggers and WordNet dictionary slightly improved translation quality.

Next comment that can be made by comparing Tables 3 and 4 is about BLEU and WordNet-BLEU metrics. We see that WordNet-BLEU score show higher match rate but it is not as high as we expected in the beginning of the experiment.

7. Conclusions

In this work, we have presented the new methodology for hybrid approach in the field of machine translation. The novelty of proposed solution is in the statistical induction of translation templates on many abstraction levels of language representation. Each language abstraction level has been received by using lingvistic resources such as POS taggers, noun and verb phrases chuckers, general purpose ontology like WordNet and bilingual dictionaries. We believe that presented models can be easily extended if new linguistic information is presented. If so, new language representations can be built and then new translation templates induced from transformed bilingual corpus. We experimented with English–Lithuanian language pair but presented method can be useful for the new languages as well. We are planing to do research on English–Russian language pair when we collect at least 10 million words corpus. English–Lithuanian language pair translations demonstrated that obtained translation results are comparable with those other language pairs if we are able to found the balance between resources available and representation abstraction levels. In his paper we haven't attributed the problem of incorrect/redundant rules due to acquisition errors or translation variety in the corpora. Some induced rules that are found to be redundant can be removed with the similar technique as in (Kenji *et al*., 2003). We hope to investigate this possibility in future research.

References

Berger, A.L., V.J. Della Pietra and S.A. Della Pietra (1996). A maximum entropy approach to natural language processing. *Computational Linguistics*, **22**(1), 39–72.

Brown, P.F., V.J. Della Pietra, S.A. Della Pietra and R.L. Mercer (1993). The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, **19**(2), 263–311.

Cunningham, H. (2002). GATE, a General Architecture for Text Engineering. *Computers and the Humanities*, **36**, 223–254.

English–Lithuanian Translation. `http://vertimas.vdu.lt`

Germann, U., M. Jahr, D. Marcu and K. Knight (2001). Fast decoding and optimal deciding for machine translation. In *Proceedings of the Conference of the Association for Computational Linguistics*. pp. 132–139.

Jurafsky, D., and J.H. Martin (2000). *Speech and Language Processing*. Prentice Hall, Englewood Cliffs, NJ.

Kenji, I., E. Sumita and Y. Matsumoto (2003). Feedback cleaning of machine translation rules using automatic evaluation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*. pp. 447–454.

Kishore P., S. Roukos, T. Ward and W.J. Zhu (2002). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the ACL*. pp. 311–318.

Koehn, F. (2005). Europarl: a parallel corpus for statistical machine translation. In *MT Summit X*.

Knight, K. (1999). Decoding complexity in word-replacement translation models. *Computational Linguistics*, **25**(4), 607–615.

Laukaitis, A., and O. Vasilecas (2007a). Asymmetric hybrid machine translation for languages with scarce resources. In *Proceedings of Eighth International Conference on Intelligent Text Processing and Computational Linguistics CICLing*. *LNCS 4394*. pp. 397–408.

Laukaitis, A., and O. Vasilecas (2007b). Integrating all stages of software development by means of natural language processing. In *Proc. of International Working Conference on Requirements Engineering*: *Foundation for Software Quality*. *LNCS 4542*. pp. 218–231.

Laukaitis, A., and O. Vasilecas (2007c). Natural language based heavy personal assistant architecture for information retrieval and presentation. In *Proc. of 12th International Conference on Human-Computer Interaction*. *LNCS 2458*. pp. 54–63.

Lithuanian Language Translation. http://www.vvam.lt/NLU/JMTranslate2.html.
Miller, G.A. (1985). WordNet: a dictionary browser. In *Proc. 1st Int'l Conf. Information in Data*. pp. 25–28.
Nagao, M. (1984). A framework of a mechanical translation between Japanese and English by analogy principle. In *Artificial and Human Intelligence*. pp. 173–180.
Official Journal of the European Union (2006). http://eur-lex.europa.eu
Och, F.J., and H. Ney (2002). Discriminative training and maximum entropy models for statistical machine translation. In *The 40th Annual Meeting of the Association for Computational Linguistics*. pp. 295–302.
Russian–English Parallel Texts (2002). http://www.homeenglish.ru/Bilingua.htm
Somers, H. (1999). Review article: example-based machine translation. *Machine Translation*, **14**, 113–157.
Vaiciunas, A., and G. Raskinis (2006). Cache-based statistical language models of English and highly inflected Lithuanian. *Informatica*, **17**(1), 111–124.

A. Laukaitis has graduated from Vilnius University Faculty of Physics in 1992. He received the PhD degree from the Institute of Mathematics and Informatics, Vilnius in 2002. He is an associated professor of the Information Systems Department of Vilnius Gediminas Technical University. His research interests include text mining, natural language interfaces, machine translation systems and knowledge management.

O. Vasilecas has graduated from Kaunas University of Technology in 1967. He received the PhD degree from Vilnius University, Vilnius in 1979. He is professor of Information System Department, and Head of the Information System Scientific Laboratory of the Vilnius Gediminas Technical University. His research interests include intelligent information system engineering, business, information and software system engineering, knowledge representation and concept modelling, business rules approach in information systems development.

Automatinio vertimo taisyklių indukcija

Algirdas LAUKAITIS, Olegas VASILECAS

Didesnė dalis pasaulio kalbų kol kas nenaudojamos automatinio vertimo sistemose. Siekiant paspartinti naujų kalbų panaudojimą automatinėse vertimo sistemose pastaruosius dešimt metų daug tyrimų buvo atlikta tam, kad būtų galima automatizuoti vertimo sistemų kūrimą panaudojant turimus išverstus tekstinius informacijos resursus. Tačiau tokie tyrimai buvo atlikti tik su taip vadinamomis aukšto dažnumo kalbomis, kuriose įmanoma gauti didelius kiekius išverstos informacijos. Lietuvių kalboje kaip ir daugelyje kitu kalbų kol kas nėra sukaupta pakankami kiekiai išverstos informacijos tam, kad būtų galima apmokyti egzistuojančias vertimo sistemas arba panaudoti sukurtas metodikas. Šiame darbe pasiūlytas ir realizuotas hibridinis metodas, kuris leidžia apmokinti vertimo sistemas iš mažesnio kiekio turimos vertimo medžiagos. Tačiau metodas reikalauja, kad sistema turėtų vertimo žodyną, pilną anglų kalbos ontologiją ir kad galėtų atlikti anglų kalbos pilną gramatinę analizę. Gauti rezultatai parodė, kad šio metodo pagalba galima sukurti vertimo sistemas, kurios pagal savo kokybę nenusileidžia pasaulyje esantiems analogams, kurie buvo sukurti statistinio modeliavimo pagalba. Esamas metodas ir sukurta programinė įranga gali būti pritaikyti ne tik lietuvių bet ir kitoms kalboms, kuriose sunku surasti didelius kiekius išverstos informacijos ir kuriose nėra resursų rankiniam vertimo sistemos kalibravimui suvedant vertimo gramatines taisykles.