SMT-BASED ASR DOMAIN ADAPTATION METHODS FOR UNDER-RESOURCED LANGUAGES: APPLICATION TO ROMANIAN

Horia Cucu[12], Andi Buzo[1], Laurent Besacier[2], Corneliu Burileanu[1]

[1]University "Politehnica" of Bucharest, Romania
{horia.cucu, andi.buzo}@upb.ro, cburileanu@messnet.pub.ro

[2]LIG, University Joseph Fourier, Grenoble, France
{horia.cucu, laurent.besacier}@imag.fr

ABSTRACT

This study investigates the possibility of using statistical machine translation to create domain-specific language resources. We propose a methodology that aims to create a domain-specific automatic speech recognition (ASR) system for a low-resourced language when in-domain text corpora are available only in a high-resourced language. Several translation scenarios (both unsupervised and semi-supervised) are used to obtain domain-specific textual data. Moreover this paper shows that a small amount of manually post-edited text is enough to develop other natural language processing systems that, in turn, can be used to automatically improve the machine translated text, leading to a significant boost in ASR performance. An in-depth analysis, to explain *why and how* the machine translated text improves the performance of the domain-specific ASR, is also made at the end of this paper. As bi-products of this core domain-adaptation methodology, this paper also presents the first large vocabulary continuous speech recognition system for Romanian, and introduces a diacritics restoration module to process the Romanian text corpora, as well as an automatic phonetization module needed to extend the Romanian pronunciation dictionary.

KEYWORDS: under-resourced languages, domain adaptation, automatic speech recognition, statistical machine translation, language modeling.

1 INTRODUCTION

In recent years, there has been an increasing interest in the field of Spoken Language Technology (SLT). For the most spoken languages of the world (Mandarin Chinese, English, Spanish, etc.) automatic speech recognition (ASR) systems, text-to-speech (TTS) systems and other natural language processing (NLP) systems are already integrated in powerful commercial spoken dialog solutions deployed on smart-phones, personal computers, call-centers, etc. Spoken dialog systems are very popular human-machine interfaces, because speech is, in fact, the most natural way to communicate and to exchange information. However, most of the languages do not possess the appropriate resources required to develop SLT systems making the development of SLT technologies very challenging.

The term *under-resourced languages* introduced by [1] refers to a language with some of (if not all) the following aspects: lack of a unique writing system or stable orthography, limited presence on the web, lack of linguistic expertise, lack of electronic resources for natural language processing (NLP) such as monolingual corpora, bilingual electronic dictionaries, pronunciation dictionaries, transcribed speech data, etc. Porting a NLP system (whether it is an ASR system, a TTS system or a complete spoken dialog system) to such a language requires techniques that go far beyond the basic re-training of the models. Indeed, processing a new language often leads to new challenges (word phonetization / segmentation / diacritization, unwritten language, etc.). The lack of resources requires, on its side, innovative data collection methodologies or models for which information is shared between languages (e.g. multilingual acoustic models [2, 3] or adaptation methods using limited data).

One important thing to note is that there are many under-resourced languages with a strong economic potential: many of them are in the top-20 of the most spoken languages of the world. This is one of the reasons why there is a growing research interest towards speech and language processing for under-resourced languages: three workshops on this topic were held in 2008, 2010 and 2012[1]. Two main challenges are addressed by many of the recent research studies: first, the challenge of solving language-specific problems and second, the challenge of optimizing costs attached to developing SLT systems.

Interestingly, under-resourced languages pose new and challenging language-specific problems that did not appear and consequently were not solved for higher-resourced languages. More over, for these languages, text data sparseness requires even more considerable efforts. This issue was addressed for the first time for two African languages: Somali [4] and Amharic [5] and one Eastern European language: Hungarian [6]. These papers address the text data sparseness by proposing word decomposition algorithms for language modeling. New phonological systems associated with some of these under-resourced languages and the difficulty to create pronunciation dictionaries is another important aspect that needs to be taken into account when developing ASR or TTS systems. This issue was recently dealt with for some under-resourced languages such as Thai [7], Amharic [8] and Vietnamese [3]. This is not only true for under-resourced languages, but the collection of textual data in a given language (and for a given domain) is also a hot topic that can be addressed using the Web as a corpus [9, 10, 11] or using machine translation systems to port text corpora from one language to another [12, 13, 14].

The cost of developing SLT systems and especially ASR systems for under-resourced languages can be successfully lowered by using various adaptation methods. Adaptation methods offer a very practical way of bootstrapping the development of ASR systems for under-resourced languages. The acquisition of speech databases and text corpora for under-resourced languages is generally a costly task, but these costs can be lowered or even avoided by using various *acoustic or language adaptation* methods. This direction has been widely explored by many recent studies which aimed at applying various adaptation techniques to under-resourced languages. We may point out that many studies have focused on acoustic modeling and fast data collection for these languages whereas fewer works were dedicated to language modeling. For example, [3] investigates several *acoustic model adaptation* techniques for bootstrapping acoustic models for Vietnamese, while [15] is concerned with *adapting acoustic models* for multi-origin non-native speakers. Several *language adaptation* methods for spoken dialog systems are proposed in [14] (English to Spanish) and [16] (French to Italian). These last two methods use statistical machine translation (SMT) to adapt language resources and models. A similar technique is used in [13] to create resources for Icelandic ASR.

This paper presents an overview of our contributions to *ASR domain-adaptation for under-resourced languages* (more specifically for Romanian). Most of our contributions are *language independent* and can be applied to any other under-resourced language, but some of them are *specific to Romanian*. Among the Romanian-specific contributions, we mention here the acquisition of the largest text corpus to be used for language modeling and the development of the first large-vocabulary ASR system for Romanian. Our language-independent contributions concern the development of an ASR domain-adaptation methodology which can be quickly used to create a domain-specific ASR system for a new language. Inspired by the work presented in [13, 14] we used statistical machine translation to create domain-specific language resources for Romanian. Going beyond the unsupervised translation scenario employed in the previous studies, we investigated the possibility of manually post-editing the machine translated text and we found that this semi-supervised scenario has a strong potential for improving the ASR performance. Moreover it is shown that a small amount of manually post-edited text, in a limited

[1] See http://www.mica.edu.vn/sltu/ for instance

domain, is sufficient to develop other NLP modules that, in turn, can be used to automatically improve the machine translated text, leading to a significant boost in ASR performance.

The work on this topic was started in spring 2011 and some of the contributions presented in this journal paper were also discussed in the following conference publications: [17], [18] and [19]. This study gives an overview of this topic and aims at describing and analyzing our methodology more widely. Among the novelties brought by this paper it is worth mentioning the in-depth analysis of the diacritics restoration system, the comparisons with previous works made for the diacritics restoration and phonetization systems and, more notable, the new SPE-based domain adaptation methodology.

The following sections are organized as follows. Section 2 of this paper is dedicated to Romanian ASR. It explains the main issues which constrained the development of large vocabulary ASR systems for Romanian and describes the state-of-the-art in this field. Section 2 also deals with some salient NLP problems in Romanian, such as diacritization and phonetization, for which two *language-independent methods* are proposed. Section 3 is the core part of this paper and describes our *contributions to under-resourced languages domain adaptation*. This section proposes several SMT-based methodologies for porting a domain-specific text corpus in a higher-resourced language (French) to an under-resourced language (Romanian), with the final goal of creating a domain-specific ASR system for Romanian. The experiments are presented in section 4. First, resource acquisition is dealt with and our baseline Romanian large vocabulary ASR is evaluated. Afterwards, our diacritization and phonetization modules are evaluated and compared with other works. Finally, our domain adaptation methodology is evaluated and the results are analyzed in order to better explain the reasons behind the effectiveness of this SMT-based adaptation methodology.

2 ROMANIAN-LANGUAGE-ASR-RELATED ISSUES

Although it is one of the European Union languages, Romanian is still considered a low-resourced language from the point of view of speech and natural language processing resources. For example, the Linguistic Data Consortium[2] (LDC) distributes speech resources for many languages of the world, but does not provide any resources for the Romanian language. The situation is similar in the case of ELRA[3] (European Language Resources Association), which also distributes language and speech resources. ELRA provides some basic linguistic resources for Romanian, but does not have any speech resources for this language. Moreover, recent work on Romanian speech recognition [20, 21, 22, 23] complain about the absence of a Romanian standard speech database and report the usage of self-created resources.

Regarding Romanian text corpora, which are needed for statistical language modeling, the situation is slightly better. LDC does not provide any standard text database for Romanian, but ELRA distributes a few small Romanian text corpora. However, [24] states that prior to their work in 2010 (which consisted in the acquisition of a 50M words Romanian corpus), there were no large, accessible, general-language corpora for Romanian. This is probably why recent works on Romanian NLP report the usage of several self-created corpora, obtained from literature books [25, 26], legal documents [27], online newspapers [28] and the web as corpus [24].

2.1 STATE-OF-THE-ART IN ROMANIAN ASR

The absence of standard speech and text resources is the main problem which inhibited the development of high-performance continuous speech recognition systems for the Romanian language. Specific speech databases have been created over the years by ASR research groups, but these resources have not been standardized or made publicly available. Due to this fact,

[2] Linguistic Data Consortium (http://www.ldc.upenn.edu)
[3] European Language Resources Association (http://www.elra.info)

large-vocabulary continuous speech recognition for Romanian was not available before this work. The latest works in speech recognition are limited to small-vocabulary tasks, basic word-loop grammars or basic n-gram language models and pseudo speaker-independency. For example, in [29] the authors report small-vocabulary (approximately 3000 words) continuous speech recognition results for a general ASR task modeled with a basic word-loop language model. The number of speakers is limited to 10. Further development and research on speech recognition algorithms and techniques is reported in [21]. This work is still limited to a small-vocabulary task (approximately 4000 words) and presents recognition results for only 11 speakers. The first study which uses more complex language models (bi-gram LMs) for Romanian is [30]. This work is also closer to speaker-independency, as it uses speech data from 30 speakers. Nevertheless this paper does not approach a general, large-vocabulary task, but a small-vocabulary (approximately 500 words), domain-specific ASR task (weather forecast).

Prior to this work, we gathered a medium-sized (64 hours) speech database which was used to create a multi-speaker Romanian acoustic model (see section 4.1.1). For language modeling we collected textual data from various on-line sources and created a general text corpus of about 169M words (see section 4.1.2). All the large Romanian news corpora were collected from the Web and lacked diacritics. Consequently, among other preprocessing operations, we were required to construct a diacritics restoration module for Romanian. This module is described in section 2.2.

An extensive phonetic dictionary of about 600k word pronunciations was also available before this work started [31]. This dictionary does not contain all the words in Romanian (more important, it does not contain any proper names). The incompleteness of this phonetic dictionary was eventually solved by creating an automatic grapheme-to-phoneme (G2P) conversion (phonetization) system. Using an automatic phonetization system is mandatory in the context of ASR adaptation because every new speech recognition task is expected to come with its own vocabulary which might comprise out-of-dictionary words. Our grapheme-to-phoneme module is described in section 2.3.

2.2 DIACRITICS RESTORATION SYSTEM FOR ROMANIAN

2.2.1 Diacritics in the Romanian language

Romanian is a language that makes intensive use of diacritics. Even though it uses only 5 diacritical characters (ă, â, î, ş and ţ), their frequency of occurrence is very high: about 30% to 40% of the words in a general text are written with diacritics. A text that lacks diacritics would generally have these characters substituted by their non-diacritical forms: a, a, i, s and t. The words in Romanian can be grouped into two categories, based on the ambiguity caused by the lack of diacritics:

a) Non-ambiguous words (words that are either written without diacritics or with a fixed diacritics pattern): *alb* (*white*, in English), *astfel (like this*), *pădure* (*forest*), *ştiinţific* (*scientific*),

b) Ambiguous words (words that can be written with several diacritics patterns): *casa* / *casă* (*the house* / *a house*), *pana* / *pană* / *până* (*the feather* / *a feather* / *until*), *bulgari / bulgări* (*Bulgarians / snow balls*), *tari* / *ţari* / *ţări* / *târî* (*strong* / *tsars* / *countries* / *to drag*).

In general, the lack of diacritics in a Romanian text can cause several problems: *reduced readability*, *apparent ambiguity* and sometimes *unsolvable ambiguity*. Although it might not seem very important, reduced readability is an important factor when the reader needs to quickly understand the message in a Romanian text. If the diacritics are missing, some sentences (especially those in which the diacritics percentage is above average) need to be read at least two times to resolve the apparent ambiguities. This is only possible when the text is large enough and the ambiguities can be resolved based on the paragraph-level context. If the text is not large

Table 1 Sentence-level unsolvable ambiguities caused by the lack of diacritics

Phrase without diacritics	Diacritics pattern #1	Diacritics pattern #2
Tancul are trei ani.	**Ţâncul** are trei ani. The boy is three years old	**Tancul** are trei ani. The tanc has three years.
Am vazut o fata frumoasa.	Am văzut o **faţă** frumoasă. We saw a beautiful face.	Am văzut o **fată** frumoasă. We saw a beautiful girl.
Romanul s-a nascut la Roma.	**Romanul** s-a nascut la Roma. The Roman was born in Rome.	**Românul** s-a nascut la Roma. The Romanian was born in Rome.
Vrem zece paturi!	Vrem zece **paturi**! We want ten beds!	Vrem zece **pături**! We want ten blankets!
Suporterii arunca cu bulgari.	Suporterii aruncă cu **bulgari**. The fans throw with Bulgarians.	Suporterii aruncă cu **bulgări**. The football fans throw snow-balls.
Sa-mi dai pana maine.	Să-mi dai **până** mâine. Give it to me until tomorrow.	Să-mi dai **pana** mâine. Give me the feather tomorrow.
Tarii au decis.	**Tarii** au decis. The powerful have decided.	**Ţarii** au decis. The tsars have decided.

enough or if only a few words from a larger text are available to the reader, then the resolution of ambiguities is more difficult or even impossible. Several such ambiguities are presented in Table 1.

Most of the Romanian news corpora which can be acquired over the web come without diacritics. In this kind of news articles the readability is significantly affected (apparent ambiguities make the texts hard to read), but unsolvable ambiguities (such as the ones in Table 1) appear rarely because the reader has access to the full paragraph-level context. However these corpora cannot be used directly to develop a Romanian ASR system due to two main reasons:

a) The ASR transcription should contain diacritics in order *to be readable*, but also *to address frequent apparent ambiguities* (which can be unsolvable if the ASR transcript is short and elliptical).

b) The ASR vocabulary is usually constructed using the most frequent words in the text corpora. The problem is that non-diacritical words cannot be modeled directly because their various diacritical forms have *different pronunciations*, have *different occurrence frequencies*, appear in *different contexts*, etc.

Therefore, an automatic diacritics restoration system is definitely needed to restore the diacritics on the text corpora used for language modeling. This approach would solve both problems stated above and allows us to create an ASR system which outputs texts with diacritics.

The restoration of diacritics on the transcription of a (diacritics-lacking) ASR system partly solves the first problem stated above, but cannot resolve the second one. The experiments presented in the following sections also support this affirmation.

2.2.2 Method description

The diacritics restoration task is regarded in this paper as a disambiguation process (such as true-casing [32]). In general, a disambiguation process aims to transforms a stream of tokens from a vocabulary V_1 to a corresponding stream of tokens from a vocabulary V_2, according to a probabilistic 1-to-many mapping. The 1-to-many mapping lists the various possibilities of transforming a word from V_1 to V_2 (plus their associated probabilities). Ambiguities in the mapping are resolved by finding the V_2 sequence with the highest posterior probability given the V_1 sequence. This probability is computed from pair wise conditional probabilities $p(V_1|V_2)$, as well as a language model for sequences over V_2.

The diacritics restoration process should transform a stream of possibly ambiguous words (in our case: words in which the diacritical characters are partly or entirely substituted by non-diacritical characters) to a corresponding stream of non-ambiguous words (in our case words with correct diacritics). Each diacritical-ambiguous word *w'* in the stream is transformed into a diacritical word estimate $\hat{w}$, given the preceding sequence of *N* diacritical words $\boldsymbol{W}$, by finding the diacritical word form w_i that maximizes this formula:

$$\hat{w} = \arg\max_{w_i} p(w_i \mid \boldsymbol{W}) \times p(w_i \mid w')$$

The first probability in the equation is given by the n-gram language model, while the second one is given by the probabilistic 1-to-many mapping. If the non-diacritical word *w'* is not found in the 1-to-many mapping, then this word form is simply copied in the stream of diacritized words ($\hat{w} = w'$).

The n-gram language model can be built using a text corpus with correctly diacritized words. The same corpus can also be used to estimate the probabilities in the 1-to-many mapping (using diacritical word counts) with the following formula (where w_j are all the diacritical forms of *w'*):

$$p(w_i \mid w') = \frac{count(w_i)}{\sum_j count(w_j)}$$

An example of a probabilistic 1-to-many mapping is shown in Table 2.

Table 2 Probabilistic 1-to-many mapping excerpt

…
dacia: dacia 1.0
fabricand: fabricând 1.0
pana: pana 0.005, pană 0.008, până 0.987
sarmana: sărmana 0.847, sărmană 0.153
tari: tari 0.047, ţari 0.002, ţări 0.942, târî 0.008
…

This diacritics restoration method is evaluated in section 4.2 in the context of ASR and also compared with other diacritical restoration methods proposed for Romanian.

2.3 GRAPHEME-TO-PHONEME CONVERSION FOR ROMANIAN

The task of automatically creating phonetic transcriptions for the words in a vocabulary is very important in speech recognition and speech synthesis and has been one hot topic in NLP research for many years. The most popular approaches proposed so far can be grouped into: rule-based approaches and machine-learning-based systems.

The rule-based approaches consider designing and applying a set of linguistic grapheme-to-phoneme conversion rules. Although these systems are most of the time very efficient, their construction requires knowledge on the language phonetics and phonology. Moreover, for some languages, the number of rules and exceptions can be huge: 1500 rules for English [33], over 600 rules for French [33], etc. On the other hand, for languages such as Spanish [34], Italian, Romanian [35], for which the pronunciation system is quite regular, the number of rules is lower and the system is simpler.

The systems that use machine learning are based on the idea that having a small set of examples of phonetic transcriptions, one can build a method that will incorporate knowledge from this training set and will be able to predict the transcription of words which are not found in this training set [33]. In practice, these systems are trained using hand built transcription dictionaries covering the most common words for that language. The most widely used systems are based on decision trees or neural networks. A more novel approach of converting graphemes to phonemes

uses statistical machine translation principles [36, 37]. The graphemes are regarded as "words" in the source language and the phonemes as "words" in the target language. A "machine translation" system is trained based on an initial phonetic dictionary and afterwards this system can be used to convert any other word to its phonetic form. Finally, another approach, to generate a pronunciation dictionary for ASR, consists in simply modeling graphemes instead of phonemes [38, 39]. The "phonetic transcription" of the word is, in fact, its written form. The graphemes are used instead of real phonemes. These systems have decent results only for languages with low grapheme-to-phoneme ambiguities.

2.3.1 Method description

In our work, we use an SMT-based approach, similar to the ones presented in [36, 37]. This type of approach has not been used before for the Romanian language. An SMT system generally translates text in a source language into text in a target language. Two components are required for training: a) a parallel corpus consisting of sentences in the source language and their corresponding sentences in the target language, and b) a language model for the target language.

For our specific task (grapheme-to-phoneme), we consider graphemes (letters) as "words" in the source language and sequences of graphemes (words) as "phrases" in the source language. As for the target language, its "words" are actually phonemes and its "phrases" are actually sequences of phonemes. Table 3 lists a few examples of lines in our training corpus.

Table 3 Examples within the phonetic dictionary (the training parallel corpus)

Example	Source language (graphemes)	Target language (phonemes)
1	d e z n o d ă m â n t u l	d e z n o d ə m ɨ n t u l
2	a c h i t â n d	a c i t ɨ n d
3	t a p i ț e r i e	t a p i ʦ e r i e

The implementation of the SMT system is based on the Moses Translation Toolkit [40]. Moses is a widely known toolkit which is mostly used for SMT tasks, but can also solve generic transduction problems as the one presented above. Our grapheme-to-phoneme translation model is trained using the default moses parameters (14 log-linear feature functions – 5 translation model scores, 1 distance-based reordering score, 6 lexicalized reordering score, 1 language model score and 1 word penalty score).

3 SMT-BASED ASR DOMAIN ADAPTATION

3.1 ASR DOMAIN ADAPTATION

ASR domain adaptation refers to the process of adjusting or changing an ASR system to make it more suitable to decode domain-specific speech utterances. Not all the components of an ASR system are subject to domain adaptation. The acoustic model, for example, is in charge of modeling the acoustic variability of the speech and is completely domain-independent. Although each domain comes with a specific vocabulary, the words are made up of the same set of phonemes regardless of the domain. On the other hand, the language model of an ASR system is in charge of modeling language variability characteristics, such as the vocabulary and the words statistical distribution, which are heavily domain-dependent.

Given the above arguments, it is clear that a general ASR system encounters great difficulties in decoding domain-specific speech utterances due to the fact its language model is not adapted to the domain. Most of the time, the general language model lacks domain-specific words and domain-specific phrases (sequences of words). These factors lead to bad objective LM performance figures such as: higher OOV rate and a lower trigram-hits percentage. Moreover,

even if the general language model contains some domain-specific words and phrases, their occurrence probabilities are small compared to the ones for the general language structures. This, in turn, leads to an increase of language model perplexity on domain-specific utterances (compared with its perplexity on general speech).

To summarize, ASR domain adaptation considers changing the general language model with a domain-specific language model and extending the vocabulary and pronunciation dictionary (to include pronunciations for the new words), while using the same acoustic model. Adapting the general language model to the domain leads to an improved prediction ability for domain-specific utterances.

A domain-specific language model can be created on domain-specific textual data (text similar to the transcriptions of the specific speech utterances). Such a domain-specific text corpus contains specific words and phrases and it is the best candidate for modeling the specificity of the domain. However, domain-specific text corpora are usually difficult to acquire. This is why domain-specific text corpora are generally small and consequently inadequate to capture both domain specificity and general language structures. Depending on the size of the domain-specific text, the language model trained on it might need to be interpolated with a general language model for improved performance. In terms of performance figures, the domain-specific language model is expected to have a better OOV-rate and trigram hits percentage (on domain-specific speech) than the general language model, thanks to the specific words and phrases within the domain-specific text corpus. The interpolation with the general language model might lead to an even better OOV-rate if the domain-specific text is too small (and does not model all the general words and phrases of the language).

3.2 SMT-BASED ASR DOMAIN ADAPTATION FOR UNDER-RESOURCED LANGUAGES

For under-resourced languages, the process of acquiring domain-specific corpora is even more difficult and more expensive. Depending on the domain, there are cases where it is not possible to collect sufficient domain-specific text for LM. This is basically the problem we are trying to solve in this paper: *is there an inexpensive way to create domain-specific text corpora and eventually domain-specific ASR systems for under-resourced languages?*

One important hint in approaching the above problem is the following: there are many textual resources (general and domain-specific) available in other high-resourced languages. The domain adaptation methodology proposed aims to exploit exactly this kind of resources to create domain-specific text for under-resourced languages. Ideally, in a *fully-supervised scenario*, these corpora could be manually translated by human experts. This process would optimize translation performance, but, on the other hand, it is too expensive and therefore not realistic. As an alternative, we propose to use statistical machine translation (SMT) to translate domain-specific data. The least expensive scenario would imply using an *already existing SMT system* in a *fully-unsupervised manner*. This means that the machine translated text (which might contain translation errors) is directly used for language modeling. Undoubtedly the errors in the translated text corpus influence the performance of the language model and eventually the ASR system. However, the results section will show that this fully-unsupervised adaptation method brings a huge WER improvement compared to the situation in which no domain adaptation is done.

An efficient balance between cost and performance can be obtained with a *semi-supervised scenario*. In this case, a human expert manually post-edits (corrects) a part of the SMT output. What is very important here is that only a small amount of post-edited sentences can bring significant improvements to the final system. Increasing the amount of post-edited sentences brings us closer to the fully-supervised scenario described above. However, we will see later in the paper that the performance curve, in function of the amount of post-edited data, shows that it becomes worthless to manually post-edit more sentences when a certain amount is reached.

The semi-supervised translation scenario described above has two disadvantages a) the manually post-editing process is costly (therefore it should be minimized) and b) the "non-post-edited" part of the corpus still contains machine translation errors. Given this, two questions arise: a) *can we minimize the post-edition effort?* and b) *is there a way to iteratively improve the quality of the machine translated text?* Even if it appears that there are two issues, both questions are closely linked because if the quality of the machine translated text improves, then post-edition effort should be smaller. Trying to answer these two questions, we decided to investigate the use of the manually post-edited part to create domain-specific MT systems.

One idea is to develop a second SMT system, specialized at translating domain-specific text. As opposed to the initial SMT system (for instance available on-line), this new SMT system can be trained using the text in the high-resourced language and the manually post-edited machine translated text. This way, the new SMT system is domain specific and might be able to better translate in-domain data.

The second idea is to use the machine translated text and the manually post-edited text to train an automatic statistical post-editor (SPE). This idea is based on the observation that the post-editing task has quite a repetitive nature. SMT systems tend to produce the same errors over and over again due to the fact that they generally produce the same output when confronted with the same input [41]. Consequently, we explored the idea of using SMT-based techniques and tools to create a SPE system that should be able to correct systematic SMT errors, based on the manual corrections observed in corpus.

It is important to note that a pre-requisite to all these methods is the availability of an initial machine translation system to port the in-domain corpus from source to target language. We believe that it is a realistic scenario since, for instance, *Google Translate* involves 63 languages[4] (including Romanian) nowadays. Moreover, the modified semi-supervised scenarios described above could also be adapted to train a domain-specific SMT system from scratch using a limited amount of manually translated data.

Although the adaptation methodology described above is applicable for any pair of languages and any specific domain, in our studies we explored the possibility of porting a tourism-specific French corpus to Romanian with the final goal of creating a tourism-specific ASR system for Romanian. The first adaptation method we developed was for the fully-unsupervised scenario. This method is presented in section 3.3.1. We continued by manually post-editing a part of the French corpus and developed four semi-supervised adaptation methods. The fully-supervised scenario was not investigated because it seemed not feasible for a real-world application (the costs of manually translating a full text corpus are quite high) and also because the improvement brought by the semi-supervised methods seemed to saturate after about 30-40% of the corpus was manually translated.

3.3 Translating the Domain-Specific Corpus - Methodology

3.3.1 Unsupervised translation scenario

The unsupervised translation scenario is the least expensive scenario among the ones presented in the previous section. This scenario implies that an already existing SMT system is used to translate the domain-specific corpus and the translation is used for language modeling without any human post-edition.

In our particular implementation we translated the domain-specific French corpus using the online Google (French-to-Romanian) machine translation system. The unsupervised domain-adaptation method implies that the machine generated translation is further used to train the domain-specific language model without any human post-edition (unsupervised). Although the

[4] http://www.techcentral.co.za/googles-babel-fish-heralds-future-of-translation/28396/

machine generated corpus contains errors, the results section will show that the domain-specific language model created using this unsupervised method is much more suitable for the domain-specific ASR task than a general language model.

3.3.2 Semi-supervised translation scenario

The semi-supervised translation scenario implies that an already existing SMT system is used to translate the domain-specific corpus and afterwards a part of this translation is manually post-edited.

In our implementation, the initial domain-specific French text was split into two parts denoted *partA* and *partB*. Both of them were Google-translated and we obtained two in-domain Romanian texts: *partA_GoMT* and *partB_GoMT*. The second, smaller part was manually corrected generating *partB_GoMTmpe*. This whole process is illustrated in Figure 1.

The first and most straight-forward use of the manually post-edited text (*partB_GoMTmpe*) was to concatenate it with the raw Google translation (*partA_GoMT*) to create a complete Romanian corpus. This corpus was further used to train a language model which obviously turned out to be better than the language model created in the fully-unsupervised scenario.

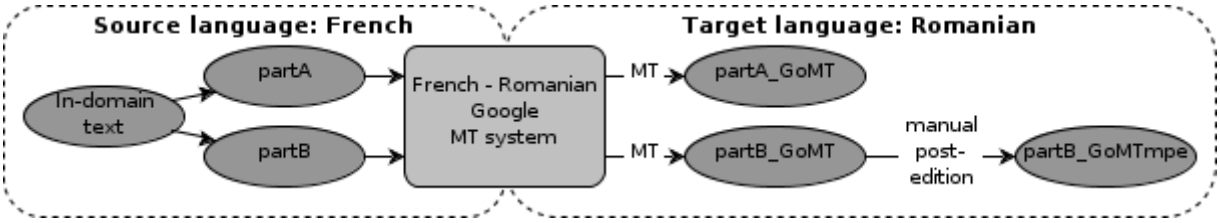

Figure 1 Semi-supervised corpus translation

Automatic improvement of the Google-translated corpus: domain-specific SMT system

As discussed in the previous section, two ideas were explored to create better translations for *partA* of the French corpus. First, we used the manually post-edited text along with its original French version to develop a domain-specific SMT system. The goal was to create a better automatic translation system to be further used (instead of Google's) to translate *partA* of the French corpus. Consequently, we regarded *partB* of the in-domain French text and the Romanian *partB_GoMTmpe* text as a parallel corpus and used it to train a domain-specific SMT system. Undoubtedly, the resulted SMT system may be worse than *Google Translate* when *partB* is very small, but it may out-perform it as more text is manually corrected. The trained SMT system was afterwards used to translate *partA* of the in-domain text, generating *partA_dsMT*. Finally, as shown in **Figure 2**, *partA_dsMT* was concatenated with *partB_GoMTmpe* to obtain a complete in-domain Romanian text.

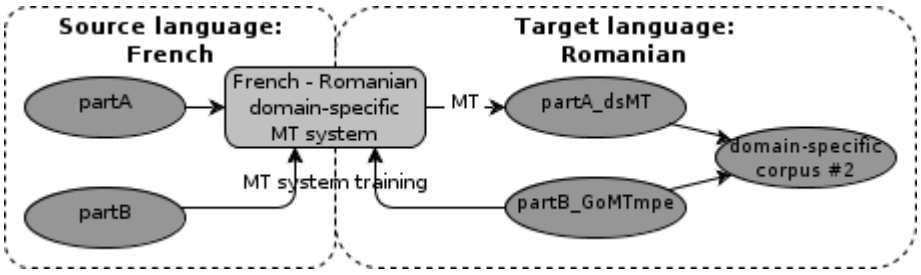

Figure 2 Second semi-supervised method: using domain specific SMT

Automatic improvement of the Google-translated corpus: SPE system

Finally, having in mind the same goal as above (to generate a better Romanian version of the French *partA*), we used *partB_GoMT* and *partB_GoMTmpe* to train a statistical post-editor (SPE). *partB_GoMT* was regarded as text in the source language (noisy Romanian) and *partB_GoMTmpe* as text in the target language (Romanian). The SPE system was built using

SMT-based techniques and tools. To summarize, the SPE "translates" incorrect text to correct text based on a phrase translation model. The SPE system was afterwards used to automatically correct *partA_GoMT* text, generating *partA_GoMTspe*. Finally, as shown in **Figure 3**, *partA_GoMTspe* was concatenated with *partB_GoMTmpe* to obtain a complete in-domain Romanian text.

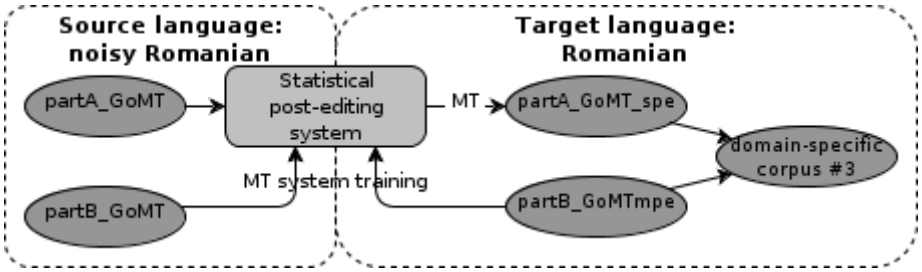

Figure 3 Third semi-supervised method: using statistical post-edition (SPE)

These three semi-supervised translation scenarios produce three complete Romanian corpora which are eventually used to train domain-specific language models. However, only *partB_GoMTmpe* is shared between the three corpora and there are three versions of *partA*: the Google-translated version (*partA_GoMT*), the version translated by the domain-specific SMT system (*partA_dsMT*) and Google-translated and statistically post-edited version (*partA_GoMTspe*). These three versions come with their own particularities. For example, the Google-translated version has a richer vocabulary than the version translated by the domain-specific SMT system, because *Google Translate* is more general and is able to potentially translate any type of sentence. On the other hand, many domain-specific phrases in *partA_dsMT* might be correct (translated similar to the phrases found in the manually post-edited training corpus) as opposed to their counterparts in the Google-translated corpus (which fails to translate domain-specific phrases).

The statistically post-edited version (*partA_GoMTspe*) benefits from the rich vocabulary of the Google-translated text, but also integrates many systematic corrections learned from the manual post-edited text.

All in all, the three Romanian versions of *partA*, and eventually the three domain-specific language models are quite different. Having now three domain-specific language models with potentially different prediction capabilities it is evident that an interpolated language model might be even better than each of its parts.

3.4 Related Work on SMT-based Domain Adaptation for ASR

Unsupervised language model domain adaptation using SMT (English to Japanese) text was proposed back in 2002 by Nakajima [12]. This paper only reports language model perplexity results, without investigating the implications on a full ASR system. Moreover, this study is limited to the basic unsupervised translation scenario and does not make any investigations on semi-supervised approaches.

In 2008, Jensson proposed a similar unsupervised language portability (English to Icelandic) method, but used it for creating the out-of-domain language model [13]. Jensson's paper [13] is also focused on the impact on ASR performance, reporting WER improvements obtained thanks to the SMT text, but the analysis is still limited to the basic unsupervised scenario.

A more recent paper [14] extends the analysis to several domains in the effort of porting an English ASR system to Spanish. The translation is also done in an unsupervised fashion.

In conclusion, the unsupervised methodology is not new, but its semi-supervised extensions, which are described above, represent a real novelty in this field. Moreover, an in-depth analysis

which investigates *why and how* the machine translated text improves the domain-specific language models is presented in the next section and represents another novel contribution.

4 EXPERIMENTAL RESULTS

4.1 DATABASES, RESOURCES AND BASELINE SYSTEMS FOR ROMANIAN ASR

4.1.1 Speech databases and acoustic models

Prior to this work, our research group gathered a medium-sized (64 hours) read speech database [53]. To the best of our knowledge, this is at the moment the largest Romanian speech database available for research purposes. It was progressively developed in a laboratory environment by recording predefined texts such as news articles, interviews, dialogs, etc. The texts were recorded by 25 speakers (11 males and 14 females). A subset of this database (5 hours of speech) was used for the ASR evaluation process (*test* part – general domain), while the rest of it was used to build the acoustic model (*train* part).

All ASR experiments presented in the following sections use the same HMM-based acoustic model. The selected HMM topology was 3 states without any skips. As voice features we used 12 Mel-Frequency Cepstral Coefficients (MFCCs) plus an energy coefficient and their first and second temporal derivates (summing up to a total of 39 features). The 36 phonemes in Romanian were contextually modeled with 4000 senones and 16 Gaussian mixtures per senone state [17]. The acoustic model was previously created and optimized (using the CMU Sphinx Toolkit[5]) with the training speech database presented above. The phonetic dictionary used in the ASR experiments was constructed as follows: a) all the vocabularies (for all the ASR tasks) were merged into a single, unified vocabulary, b) all the words within the unified vocabulary, which were found in our pre-existing phonetic dictionary, were transcribed using this dictionary, c) all the other words were transcribed using the grapheme-to-phoneme conversion system presented in section 2.3.

For the tourism-specific ASR task, the evaluation speech database (*test* part– domain specific) was obtained as follows: 300 sentences were randomly selected out of a French tourism-specific corpus (see section 4.1.2), and then manually translated to Romanian and recorded by three speakers. The size of the evaluation database is about 55 minutes (900 utterances). Obviously, the 300 sentences were removed from the French tourism-specific corpus before it was further used for language model training.

4.1.2 Text corpora

This section describes the various text corpora used in our experiments. Note that most of the corpora and especially the larger ones have been acquired using the Web.

The French *media* corpus [42] which comprises tourism specific transcriptions of spontaneous French speech was available prior to these experiments. For the domain adaptation experiments, this corpus was machine translated to Romanian using the Google on-line translation system[6]. This machine translated corpus is called *mediaMT* in the rest of this paper and it consists of about 10k sentences (64k words). A subset (4k sentences) of *mediaMT* was manually post-edited (the machine translation errors were corrected) giving birth to *mediaPE*. In order to estimate the performance of the Google French-Romanian translation system for our translation task, we compared the post-edited corpus (*mediaPE*) as reference and the corresponding subset of the Google translation (*mediaMT*) as hypothesis. The resulted BLEU score [47] was 40.87.

[5] The CMU-Sphinx Speech Recognition Toolkit (http://cmusphinx.sourceforge.net)

[6] Google on-line machine translation system (http://translate.google.com) as of May 2011

The *europarl* corpus is a free corpus available on-line[7] [43] for all the European Union's languages and comprises the discussions in the European Parliament. The English or French *europarl* corpora are larger as these countries were part of the EU from the beginning, while the Romanian corpus consists of 225k sentences only (5.3M words with correct diacritics).

The *9am* and *hotnews* corpora were obtained by automatically downloading and preprocessing all articles from two on-line[8] newspapers. The texts address all types of news. The *9am* corpus consists of 3.5M sentences (63M non-diacritized words) and the *hotnews* corpus consists of 6M sentences (100M non-diacritized words).

Finally, the *misc* corpus [26] was already available before this study. It comprises newspaper articles, PhD theses, and literature. It consists of 400k sentences (11M words with correct diacritics).

All the information regarding these text corpora is summarized in Table 4. The diacritics column shows that the larger news corpora were not originally diacritized. For these texts, the diacritics were restored using a system that was specifically developed for this purpose (see section 2.2).

Table 4 Created/Acquired Romanian Text Corpora

Corpus name	Domain	Sentences	Words	Diacritics
mediaMT	tourism dialogs	10k	64k	available
mediaPE	tourism dialogs	4k	27k	available
europarl	EU discussions	225k	5.3M	available
9am	news	3.5M	63M	restored
hotnews	news	6.0M	100M	restored
misc	journal, literature, other	400k	11M	available

These text corpora were further used to create the n-gram language models required by various experiments. The *europarl*, *9am* and *hotnews* corpora were selected to be representative for the Romanian language and were used to train a general language model for Romanian. The *mediaMT* and *mediaPE* corpora were solely used in the domain adaptation experiments presented in section 4.4. Finally, the originally diacritized corpora (*europarl* and *misc*) were used to train and evaluate the diacritics restoration system (see section 4.2).

4.1.3 Baseline for Romanian ASR

The HMM-based acoustic model and the unified phonetic dictionary described in section 4.1.1 together with a general language model created using the *europarl*, *9am* and *hotnews* corpora were used to create our first Romanian large-vocabulary continuous speech recognition system.

The general language model is a tri-gram, closed-vocabulary language model and was created using the SRI-LM toolkit[9]. As a smoothing method we used the Good-Turing discount method (the default in SRI-LM). The vocabulary size was limited to the most frequent 64k due to the ASR decoder (Sphinx3) limitation.

The performance of the system was assessed on our two test sets (general and specific domains). The results are presented in Table 5. It is worth noting the high OOV rate obtained on the domain-specific test set (one of the reasons for the high WER on this test set). The perplexity comparison is not relevant here. Perplexity is computed only for in-vocabulary words and therefore the comparison is relevant only for experiments with similar OOV rates.

---

[7] European Parliament Proceedings Parallel Corpus (http://www.statmt.org/europarl)

[8] 9am (http://www.9am.ro) and Hotnews (http://www.hotnews.ro)

[9] The SRI Language Modeling Toolkit (http://www-speech.sri.com/projects/srilm)

Table 5 Baseline Romanian ASR systems

		General speech			Domain-specific speech		
Acoustic model	**Language model**	**PPL**	**OOV**	**WER**	**PPL**	**OOV**	**WER**
HMM acoustic model (4000 senones, 16GMMs)	general language model (64k words) from europarl+9am+hotnews	183.2	1.8%	20.4%	164.7	4.3%	29.7%

To the best of our knowledge, these are the first large-vocabulary (>60k words) continuous speech recognition results reported for the Romanian language.

4.2 DIACRITICS RESTORATION EXPERIMENTS

4.2.1 Experimental setup and results

The two corpora which originally comprised words with correct diacritics (*europarl* and *misc*) were used to build up the diacritics restoration system (the n-gram language model and the 1-to-many mapping). In fact, the *misc* corpus was beforehand split into a larger part (90%) entirely used for training and a smaller part (10%) to be used in the evaluation process.

In order to find the best setup for the diacritics restoration system, we varied the LM order N from 2 to 5 and we also tried to use a plain 1-to-many mapping (in which all surface forms – with or without diacritics - of a non-diacritized word have equal likelihoods). The SRI-LM Toolkit was used to create the language models and also to perform the disambiguation (*disambig* tool) of an input (non-diacritized) sentence.

The various versions of the system were evaluated in terms of word error rate (WER) and character error rate (ChER), on the evaluation part of the *misc* corpus, using the NIST Scoring Toolkit *sclite*[10]. The experimental results are presented in Table 6.

Table 6 Diacritics restoration results

Exp	**LM**	**1-to-many map**	**WER**	**ChER**
1	2-gram	Probabilistic	2.07%	0.50%
2	3-gram	Probabilistic	1.99%	0.48%
3	4-gram	Probabilistic	1.99%	0.48%
4	5-gram	Probabilistic	2.00%	0.49%
5	3-gram	Plain	2.24%	0.54%

As noted from Table 6 the LM-order (*N*) variations do not bring important improvements if the training corpus is relatively small (~15M words, in our case). However, the results obtained using a probabilistic 1-to-many mapping, instead of a plain 1-to-many mapping, are better.

4.2.2 Comparison to previous works for Romanian

Besides this method, several fundamentally different diacritics restoration methods were developed for the Romanian language.

A more elaborate *knowledge-based diacritics restoration method* using *part-of-speech (POS) tagging* to disambiguate the different diacritical words hypotheses, was introduced in [44] and refined in [45]. Nevertheless, this method was reported to have slightly lower performance figures than our proposed algorithm: a 2.25% WER and a 0.60% ChER. These results were obtained on a different test set than ours (there is no standard evaluation corpus for Romanian diacritics restoration).

[10] NIST Speech Recognition Scoring Toolkit (http://www.nist.gov/speech/tools)

In [26] the diacritics restoration system is regarded as a *sequential filtering process* based on *unigrams and bigrams of diacritical words* and *trigrams of diacritical word-suffixes*. This method needs only a medium size text corpus to train the various language models and to create a 1-to-many mapping connecting non-diacritical word forms to all their diacritical word forms. The authors insist on the fact that this method is adapted to Romanian thanks to the usage of *word-suffixes trigrams*. In 2008, the authors reported a 2.13% WER (on the same test set as ours), but after various refinements [46] they reported even better results: a 1.4% WER and a 0.4% ChER (although on a different test set).

In conclusion, we assert that the diacritics restoration system we have developed is one of the best available for Romanian, and can be considered as sufficient for our ASR experiments.

4.2.3 Deeper analysis of our system

In general, the evaluation of diacritics restoration is made in terms of WER and ChER. However, these two performance figures are unable to highlight the system's capability to restore certain diacritical characters individually. A more in-depth, character-based analysis has to be performed in order to obtain more details to know which of the diacritical characters are better restored and which of them are more error-prone.

For this evaluation, we used three other performance figures: *precision*, *recall* and *F-measure*. Precision is the ratio between the number of correctly inserted diacritics and the number of diacritics in the hypothesis text, while recall is the ratio between the number of correctly inserted diacritics and the number of diacritics in the reference text. F-measure is the harmonic mean of precision and recall.

Table 7 lists the various performance metrics for the individual characters that are subject to diacritics restoration. Based on these results, the first conclusion that can be drawn is that the method exhibits better performance metrics for the non-diacritical characters (*a, i, s, t*). Also, some ambiguity classes (*i* / *î*; *s* / *ş*) are almost perfectly solved, while others (*a* / *ă* / *â*) pose serious problems. The *a* / *ă* ambiguity is a specific and difficult problem for Romanian, because all the feminine nouns and adjectives whose singular, indefinite form ends in *ă* have their singular, definite forms ending in *a*. Consequently, these word forms cannot be disambiguated easily and we would probably need higher order n-gram models or some linguistic knowledge-based method to approach this ambiguity. The individual character results and the above conclusions are consistent with the ones presented in [26].

Table 7 Diacritics restoration: individual character evaluation

Ambiguity class	Char	Precision	Recall	F-measure
a / ă / â	a	98.28%	97.71%	97.99%
	ă	94.42%	96.13%	95.27%
	â	98.79%	97.55%	98.16%
i / î	i	99.97%	99.88%	99.92%
	î	99.26%	99.65%	99.45%
s / ş	s	99.75%	99.62%	99.69%
	ş	98.71%	99.14%	98.92%
t / ţ	t	99.52%	99.62%	99.57%
	ţ	97.74%	97.21%	97.47%
all	all	98.73%	98.73%	98.73%

4.2.4 Diacritics restoration in the context of ASR

The reason why we developed the diacritics restoration system was to correct the large text corpora we collected from the Web (*9am* and *hotnews*), with the final goal of creating a general language model for our Romanian ASR system. Consequently, the goal of this section is to evaluate the diacritics restoration system in the context of ASR.

For this experiment, we used the HMM-based acoustic model and the unified phonetic dictionary described in section 4.1.1. For language modeling we used the general news corpora: *9am* and *hotnews*. In *exp 1* and *exp 2*, the language models were trained on the original, non-diacritized corpora. In *exp 3* the language model was trained on the text with diacritics restored using our method, while in *exp 4* the language model was trained on the text with diacritics restored using the system of [46] who agreed to apply their method to our corpora. In *exp 2* we restored the diacritics on the ASR output. Note that the ASR evaluation was done on the domain-specific speech database for which the general language model (trained with the news corpora) is not very well suited.

Table 8 presents the results in terms of ASR word error rate (WER). In *exp 1* we compared hypotheses texts without diacritics with reference texts with diacritics. The high WER argues for the need of diacritics restoration for Romanian. Comparing *exp 2* and *exp 3* we can conclude that better results are obtained if the diacritics restoration is done on the text corpus, before language modeling, as opposed to the hypotheses texts. Experiments 3 and 4 show the difference in ASR performance between the best diacritics restoration system for Romanian [46] and the method we have developed.

Table 8 Diacritics restoration in the context of ASR

Exp	Diacritics restoration	WER
1	no diacritics restoration	64.5%
2	on hypotheses text, after ASR (using this method)	30.5%
3	on text corpus, before LM (using this method)	29.7%
4	on text corpus, before LM (using [46])	29.4%

4.3 GRAPHEME-TO-PHONEME CONVERSION EXPERIMENTS

4.3.1 Experimental setup and results

The already available phonetic dictionary described in section 4.1.1 is considered as our "parallel corpus" needed for SMT training. It was randomly split into three parts: a) a training part (580k words), b) an optimization (tuning) part (10k words) and c) an evaluation part (10k words). The "target" part of this dictionary (the phonetic sequence associated to a word) served as a training corpus for target language model training.

The translation model's optimization could have been made by minimizing the phone error rate (PhER), but we used two already available tuning methods: a) maximization of the BLEU score [47] (the default metric largely used in the machine translation community) and b) minimization of the position independent phone error rate (called PIPhER in this paper) which is directly inspired from the position independent error rate (PER) metric presented in [48]. The evaluation of the grapheme-to-phoneme (seen as a translation task) performance is made using BLEU score, phone error rate (PhER) and word error rate (WER). The results are summarized in Table 8. The most relevant evaluation criterion in the table is the phone error rate. We did not optimize the translation model using this criterion due to the fact that a mertPhER module was not available at that point.

Table 9 SMT-based grapheme-to-phoneme conversion performance

Exp	Optimization	BLEU	PhER	WER
1	None	98.89	0.53%	4.79%
2	BLEU	99.49	0.33%	3.24%
3	PIPhER	99.39	0.31%	2.76%

Note that in the above table BLEU score is indicated for information only, since it is the default evaluation metric for machine translation. We are aware that it is not suitable for a grapheme-to-phoneme conversion task.

4.3.2 Comparison with previous works for Romanian

Over the past decade, several research groups have created grapheme-to-phoneme tools for the Romanian language. These tools are regarded as essential modules within text-to-speech systems [49, 50, 51, 46] or for the generation of phonetic dictionaries [35, 52]. The main methodologies utilized are still the ones used for other languages: machine learning [49, 52], rule-based [35, 46] and hybrid (machine learning and conversion rules) [50, 51]. All these papers report evaluation results in terms of word error rate (WER) or phone error rate (PhER). Even if the results reported in the above papers are not directly comparable due to the different experimental setups (different set of phonemes, different evaluation words, different number of evaluation words, etc.) and the lack of complete evaluation metrics (PhER and WER), we have summarized them in Table 10.

Table 10 Grapheme-to-phoneme conversion tools for the Romanian language

System	# Evaluation words	PhER	WER
[49]	1000	n/a	2.9%
[50]	400	n/a	~ 5%
[51]	1000	n/a	5.2%
[35]	4779	0.72%	4.79%
	15599	n/a	9.46%
[52]	100	7.17%	n/a
[46]	11819	n/a	3.01%
our method	10000	0.31%	2.76%

4.3.3 Discussion

The grapheme-to-phoneme system created using a SMT-based approach provides state-of-the-art results (Table 10). Even if the test sets are different, our large (10k words) evaluation database assures us that our method performs similar to the state-of-the-art G2P systems previously developed for Romanian.

Such a grapheme-to-phoneme system allowed us to be able to constantly update the phonetic dictionary for a new ASR task, which is a mandatory feature for ASR domain adaptation. The words in the task specific vocabulary need to be phonetically transcribed before the actual recognition process can be started. This process is performed in two steps:

a) All the words within the specific vocabulary which are found in the 600k words phonetic dictionary are transcribed using the entries in the 600k dictionary,
b) All the other words are transcribed using our grapheme-to-phoneme system.

4.4 ASR DOMAIN ADAPTATION EXPERIMENTS

4.4.1 SMT and SPE systems

The semi-supervised domain adaptation methods presented in section 3.3.2 require the development of a domain-specific SMT system (for the second method) and a domain-specific SPE system (for the third method). The main difference between a SMT and a SPE is that the first translates text from one language to another, while the second corrects machine translation output (can be seen as a translation from a noisy input to a clean output in the same language). The development of the two systems is similar and was performed using the Moses Toolkit.

In the case of the domain-specific SMT system, the second part of the French domain-specific corpus (*partB*) and its Google-translated and post-edited counterpart (*partB_GoMTmpe*) were regarded as parallel corpora and were used for training. The manually post-edited text was also used to create a domain-specific language model (also needed to train the SMT system).

In the case of the domain-specific SPE system, the Google-translated part of the French domain-specific corpus (*partB_GoMT*) and its manually post-edited version (*partB_GoMTmpe*) were regarded as parallel corpora and were used for training. The manually post-edited text was again used to create the domain-specific language model (also needed to train the SPE system).

In both cases, the size of the training corpus varied from 500 sentences (5% of the domain-specific corpus) to 4000 sentences (40% of the domain-specific corpus). We did not use any of the text data for optimization or translation evaluation. The methods and consequently the translation/post-editing systems were evaluated only in the context of ASR adaptation (see experiments in section 4.4.3).

4.4.2 Experimental setup

For the domain adaptation experiments, various language models were trained using the general corpora: *europarl*, *9am* and *hotnews* and the tourism-specific corpora: *mediaMT* and *mediaPE* (see section 4.1.2 for more details). All these language models are tri-gram, closed-vocabulary language models and were created using the SRI-LM toolkit (smoothing method being Good-Turing discounting). This toolkit was also used for the interpolation of the general language model with the various domain-specific language models. The interpolation was systematically done with the weights 0.1 for the general LM and 0.9 for the domain-specific LM. The tuning of the interpolation weights was not considered at the moment (so our models were evaluated with equivalent weighting conditions). For some of the language models, the number of unigrams had to be limited to the most frequent 64k due to the ASR decoder (Sphinx3) limitation.

The evaluation of all the language models was done in terms of perplexity (PPL), out-of-vocabulary (OOV) rate, percentage of trigram hits and ASR word error rate (WER) on the tourism-specific speech database. The HMM-based acoustic model and the unified phonetic dictionary used in all the ASR experiments, as well as the evaluation speech database, are those described in section 4.1.1.

4.4.3 Unsupervised adaptation results

The unsupervised domain adaptation method is evaluated first and the results are presented in Table 11. We see that the unsupervised adaptation method produces a domain-specific language model (*exp 2*) which is significantly better than the general language model. In other words, the Google-translated domain-specific corpus is much better than the general Romanian corpus on the domain-specific task. The interpolation of these two language models issues an even better language model (*exp 3*).

Table 11 Unsupervised adaptation results - evaluation made on domain-specific speech

Exp	Language model	PPL	OOV	3gram hits	WER
1	general LM	164.7	4.27%	51.0%	29.7%
2	domain-specific LM	40.8	3.15%	31.1%	18.7%
3	domain-specific LM interpolated with general LM	42.5	0.80%	55.4%	16.2%

The general language model and the domain-specific language model have relatively high out-of-vocabulary rates (4.27%, respectively 3.15%). On the other hand, the interpolated language model has a very small out-of-vocabulary rate (0.80%). We can conclude that the general Romanian corpus lacks some domain-specific words, while the domain-specific translated-corpus lacks some general Romanian words. Nevertheless, the two corpora complement each other and the interpolated language model benefits from both vocabularies.

Another interesting discussion regards the perplexity and the trigram hits for the three language models. The general language model has a large perplexity on the test set (it *is not* suitable to predict tourism-specific phrases), although 51.0% of the words are full trigram hits. This means that more than half of the 3-word sequences in the test corpus appeared in the training corpus. On the contrary, the domain-specific language model exhibits a better perplexity on the test set (it *is* suitable to predict tourism-specific phrases), although only 31.1% of the 3-word sequences in the test corpus appeared in the training corpus. Its better perplexity comes from higher prediction probabilities assigned to domain-specific unigrams and bigrams.

The interpolated language model benefits from the low perplexity of the domain-specific language model and the high trigram hits of the general language model. Thanks to this, its speech recognition results are the best. In conclusion the machine translated corpus plays a significant role in the improvement of the general language model and consequently the improvement of ASR system for a domain-specific task.

4.4.4 Semi-supervised adaptation results

Table 12 presents the ASR results obtained for the domain-specific language models before the interpolation with the general language model. The results obtained for the three semi-supervised methods (see section 3.3.2) are grouped in three main columns. The results obtained for the unsupervised adaptation method are also repeated in this table (the first results line on every column), because they represent the baseline for all the other experiments. The subsequent lines in the table show the semi-supervised methods improvements obtained over the baseline as 5%, 10%,… 40% of the Google translated corpus was manually post-edited.

Statistical confidence for the results

The WER values presented in Table 12 (and the subsequent tables) are in some cases very similar and therefore need to be supported with some statistical confidence measure. To compute confidence intervals for the results in these tables, the experiments for method 1, for partB size of 5% (second line in Table 12) were repeated 8 times. In each experiment a different 5% part of the Google translated corpus was selected for manual post-editing. We obtained an average WER of 15.3 and a 95% confidence interval of ±0.30 (from 15.0 to 15.6).

The most important conclusions can be drawn given the results in Table 12 and their statistical relevance are summarized below.

A small amount of post-edited data is enough to train a specific SPE system, but not a specific SMT system

First, we observe that even when a small amount of data (such as 5% of the Google translated text) is manually post-edited, all the performance figures are significantly better for the first-

Table 12 Domain-specific language model results (before interpolation with general LM) - evaluation made on domain-specific speech

		method 1 General MT (GoMT)					method 2 Specific SMT (dsMT)					method 3 Specific SPE (GoMTspe)			
partB size		**PPL**	**OOV [%]**	**3gram hits [%]**	**WER [%]**		**PPL**	**OOV [%]**	**3gram hits [%]**	**WER [%]**		**PPL**	**OOV [%]**	**3gram hits [%]**	**WER [%]**
00%		40.8	3.15	31.1	18.7		40.8	3.15	31.1	18.7		40.8	3.15	31.1	18.7
05%		34.8	2.08	34.0	15.1		31.8	6.68	35.3	22.0		32.7	2.40	35.9	15.3
10%		32.5	1.76	35.2	14.6		28.4	3.95	38.4	17.4		28.9	1.92	38.9	13.6
20%		28.7	1.50	37.9	13.0		25.3	2.88	41.2	15.4		26.6	1.60	40.4	13.0
30%		26.3	1.39	39.4	12.7		23.6	2.30	42.1	14.2		24.9	1.50	41.3	12.5
40%		24.8	1.39	41.3	12.5		23.5	1.98	42.7	13.6		24.7	1.39	42.9	12.5

method system. This is a very important observation, because it shows that a small amount of extra work (manual post-edition of 500 sentences) can lead to a significant improvement over the baseline.

On the other hand, the second-method system that uses these 5% (of the Google translated text) displays a significantly higher WER (line 2 compared to line 1). Even if the trigram hits and perplexity are better, the out-of-vocabulary rate is much worse and it causes the higher WER. This happens because these 5% (500 sentences) are not enough to train a decent SMT system; many words cannot be translated by this system, resulting in a pseudo-Romanian domain-specific corpus, which is clearly not suited for language modeling and ASR. The second-method system needs at least 1000 manually post-edited sentences to outperform the baseline. Note that even with a higher OOV rate (3.95%), the second-method system obtains better ASR results than the baseline (17.4% WER compared to 18.7% WER) thanks to a better perplexity (28.4).

Interestingly, although the 500 manually post-edited sentences are not enough to train a domain-specific SMT system, they are very useful in case they are used to train a SPE system. This affirmation is sustained by the third-method system results (line 1 in the table). 500 manually post-edited sentences are sufficient for the third-method system to clearly outperform the baseline results (15.3% WER compared to 18.7% WER). The third-method system shows similar results to the first-method system; a significant performance difference was observed only for 1000 manually post-edited sentences (see line 3 in the table).

The ASR improvements brought by the semi-supervised methods do not increase linearly with the amount of post-edited data

A second important conclusion is that all the semi-supervised methods issue better and better ASR systems as more machine translated sentences are being post-processed (the only exception is the one discussed above for the second-method systems). The growth in performance is not linear and appears to saturate when 30% to 40% of the machine translated data is post-edited.

The Google translation has a richer vocabulary than the texts created by our SMT/SPE systems

also shows that, when the same amount of data is manually post-edited, the first-method systems have systematically better OOV-rates. This means that from the vocabulary point of view the Google-translated part of the French corpus (*partA_GoMT*) is better than its statistically post-edited version (*partA_GoMTspe*) created by third-method system and also better than the translation created by our domain-specific SMT system (*partA_dsMT*). The language model trained with this latter corpus (*partA_dsMT*) has the poorest OOV-rate because the domain-specific SMT systems can only translate the words found in the small training corpus (*partB – partB_GoMTmpe*), leaving the other words in their "French version". As it will be shown later, the poor-vocabulary disadvantage of the latter two corpora can be surmounted by interpolating the domain-specific language models with a general language model (which comes with rich and general vocabulary).

Table 13 Improved domain-specific language model results (after interpolation with general LM) - evaluation made on domain-specific speech

		method 1 General MT (GoMT)					method 2 Specific SMT (dsMT)					method 3 Specific SPE (GoMTspe)			
partB size		**PPL**	**OOV [%]**	**3gram hits [%]**	**WER [%]**		**PPL**	**OOV [%]**	**3gram hits [%]**	**WER [%]**		**PPL**	**OOV [%]**	**3gram hits [%]**	**WER [%]**
00%		42.5	0.80	55.4	16.2		42.5	0.80	55.4	16.2		42.5	0.80	55.4	16.2
05%		34.4	0.80	56.0	14.6		36.3	0.80	58.8	14.2		31.7	0.80	57.5	13.6
10%		32.4	0.53	56.8	13.9		30.1	0.53	58.6	12.7		28.3	0.53	58.4	12.9
20%		29.0	0.48	57.7	13.1		26.7	0.48	59.5	12.6		26.2	0.48	59.3	11.9
30%		26.6	0.48	58.2	12.4		24.3	0.48	59.9	11.6		24.7	0.48	59.4	11.9
40%		25.2	0.48	59.1	12.2		23.8	0.48	60.2	11.5		24.4	0.48	60.1	11.7

The SMT/SPE systems produce new and useful trigrams

Comparing the three semi-supervised methods, we observe that, when the same amount of data is manually post-edited, the second- and third-method systems systematically display better trigram hits than the first-method systems. This means that the newly developed SMT/SPE systems produce translations which include some new and useful trigrams. The trigram hits analysis is detailed in section 4.4.6.

Table 13 presents the ASR results obtained for the domain-specific language models *after* the interpolation with the general language model. The results obtained for the three semi-supervised methods are again grouped in the three main columns and the unsupervised method results are again repeated on the first line of the table as they represent the baseline. The results show that after interpolation all systems display the same trend of lower WERs as more manually post-edited sentences are used, just as before interpolation. The growth in performance is again non-linear and appears to saturate when 30% to 40% of the machine translated data is manually post-edited.

The lack of coverage which characterized the SMT/SPE corpora was overcome

After interpolating the domain-specific language models with the general LM, the OOV rates are equal for all the systems (when the same amount of data is manually post-edited). This means that the lack of coverage which characterized the SMT-generated and SPE-generated corpora before interpolation (

) was overcome. Consequently, the second- and third-method systems continue to be better in terms of perplexity and trigrams hits, but now outperform the first-method system in terms of WER (thanks to the general language model which reduced the OOV rates).

The interpolation with the general LM leads to more similar performance figures for the systems

Comparing Table 12 and Table 13, we easily observe that after interpolation the systems performance figures are more similar than before interpolation. Regardless, we must remark the SPE system's efficiency when only a small amount of data is manually post-edited (line 1 in the table: 13.6% WER).

The interpolation with the general LM improves the OOV rates and the trigram hits

Comparing the corresponding lines in Table 12 and Table 13, we may conclude that, after interpolation, the OOV rates and the trigram hits are much better. This is why the WERs are also much lower for these ASR systems. This improvement in vocabulary and trigrams coverage seems to be essential and is brought by the general language model.

Table 14 Mixed semi-supervised adaptation results - evaluation made on domain-specific speech
left: before interpolation with general LM

		mixed method			
partB size		**PPL**	**OOV [%]**	**3gram hits [%]**	**WER [%]**
00%		40.8	3.15	31.1	18.7
05%		28.0	2.03	42.5	14.1
10%		26.1	1.71	43.6	13.6
20%		24.0	1.50	44.8	12.6
30%		22.8	1.39	45.1	12.2
40%		22.5	1.39	45.6	12.2

right: after interpolation with general LM

		mixed method			
partB size		**PPL**	**OOV [%]**	**3gram hits [%]**	**WER [%]**
00%		42.5	0.80	55.4	16.2
05%		28.3	0.75	59.7	12.7
10%		26.4	0.48	59.6	12.4
20%		24.4	0.48	60.3	12.0
30%		23.1	0.48	60.6	11.4
40%		22.9	0.48	60.7	11.3

4.4.5 Mixed adaptation results

The three semi-supervised methods evaluated above use different Romanian versions of *partA* of the French corpus. As discussed above each version of the text, either if it is translated using Google (*partA_GoMT*), or if it is translated using the newly developed SMT system (*partA_dsMT*), or if it is the statistically post-edited version of the Google translation (*partA_GoMTspe*), has its own advantages and disadvantages. Therefore a good idea may be to use all these corpora to create a domain-specific language model. Table 14 presents the results obtained by this mixed language model before and after the interpolation with the general LM.

If we first compare Table 14 with Table 12 and Table 13 we observe that all the performance figures for the mixed-method system are better than those for the single method systems, regardless of the amount of manually post-edited data. In particular, for small amounts of corrected data (5-10%), the mixed-method system is significantly better than the others. Thanks to its construction methodology, this system benefits from in-domain words and sequences of words from the Google translation, the domain-specific translation and the statistically post-edited Google translation. This explains the better perplexity, out-of-vocabulary rate, and 3-gram hits and consequently the lower WER.

When a large amount (30-40%) of corrected data is used, the mixed-method system has similar performance figures as the single-method systems. In conclusion, the mixed semi-supervised domain adaptation method is recommended when only a small part of corrected text is available.

4.4.6 Why is this working? In-depth n-gram hits analysis

As shown in the previous sections, the improved domain-specific language models have a good ability to predict (55% to 60% trigram hits) both domain-specific word sequences and out-of-domain word sequences (thanks to the interpolation with a general language model). In this study, the general language model was the same for all experiments, so, if we want to answer the question *why and how the proposed methodologies bring improvements in ASR?*, we have to analyze the various domain-specific language models before interpolation. Table 13 showed the results for all these language models. The language models on the last line were selected and analyzed in Table 15 from the point of view of their ability to predict specific words.

Table 15 shows seven trigram examples and analyses the way the language models manage to predict the bolded word in the given context. *3-gram* means the language model was able to predict the bolded word in the given trigram context and *2-gram* means the language model needed to back-off to bigrams to predict the bolded word. *1-gram* means the language model needed to back-off to unigrams to predict the bolded word and *OOV* asserts the LM cannot predict the bolded word (it is out-of-vocabulary).

Table 15 N-gram hits for the general and domain-specific language models (examples)

		Trigram examples							
		a	**b**	**b, d**	**C, d**	**c**	**c, d, e**	**f**	**Type**
		o cameră **single**	care acceptă **animale**	într-o locație **liniștită**	puteți să-mi **dați**	și acum **pentru**	prea scumpă **într-un**	nopți pentru **Belfort**	**Ro text**
Language model	**3-gram hits**	a **single** room	which accepts **animals**	in a **quiet** place	Can you **give** me	and now **for**	too expensive **in a**	nights at **Belfort**	**En text**
general LM	51.0%	3-gram	1-gram	1-gram	3-gram	3-gram	2-gram	OOV	
domain-specific LM (unsupervised method)	31.1%	3-gram	3-gram	2-gram	1-gram	1-gram	1-gram	1-gram	
domain-specific LM (semi-supervised method 1)	41.3%	3-gram	3-gram	3-gram	3-gram	1-gram	2-gram	1-gram	
domain-specific LM (semi-supervised method 2)	42.7%	3-gram	3-gram	3-gram	3-gram	1-gram	3-gram	1-gram	
domain-specific LM (semi-supervised method 3)	42.9%	3-gram	3-gram	3-gram	3-gram	1-gram	2-gram	1-gram	

Note that there are trigrams which can be very well predicted by all the analyzed language models (type a), but also trigrams that can only be predicted by the domain-specific language models (type b). The importance of the interpolation with the broader, general language model is motivated by its higher trigram hits (51%) and by trigrams which can only be well predicted by it (type c). The benefit brought by the semi-supervised methods is revealed by examples of type d. Only a few trigrams can be better predicted by the second-method systems, when compared to the first-method systems (see the small difference in trigram hits and examples of type e). There are also a few cases of trigrams which can be better predicted by the third-method systems, when compared to the second-method systems, but these are not shown in the table due to the lack of space. Finally, there are examples of trigrams which cannot be well predicted by any of the analyzed language models (type f). The frequency of occurrence for these six types is difficult to estimate, but the big picture is illustrated by the trigram hits column.

5 CONCLUSION

In this study we proposed several domain-adaptation methods that can be used to develop a domain-specific ASR system for a low-resourced language. One of the most significant findings to emerge from this study is that statistical machine translation can be very useful in creating domain-specific language resources (that can be further used for language modeling and automatic speech recognition).

We investigated the potential of unsupervised translation, as being the least expensive adaptation method and demonstrated that the domain-specific language model trained in this manner outperforms the general language model by 45% relative (in terms of automatic speech recognition WER). Going further we explored the importance of manually post-editing the machine translated text and showed that the improvements over the baseline (the unsupervised translation scenario) are between 10% and 25% relative, depending on the amount of corrected sentences (500 to 4000).

We developed the semi-supervised methodology even further and used the manually post-edited text to train two automatic systems that were used to improve the quality of the machine translated text. The first one was a domain-specific SMT system and the second one was a statistical post-edition (SPE) system. Finally this paper reports on mixing all the domain-specific text corpora resulted from various methods taking advantage of the fact that each one brings a particular plus. The results show that this final methodology requires way less manually post-edited text to obtain the same performance, thus drastically decreasing the cost of development.

Besides the domain-adaptation methodology which is the core contribution of our work, this paper also evaluated the first large vocabulary ASR system for Romanian and discussed the

most important language-specific issues that were surmounted during the development process. A new diacritics restoration system was proposed, evaluated and compared with other works on Romanian. Eventually this system was used to process the large Romanian corpora which were collected for language modeling (and lack diacritics). Moreover, a grapheme-to-phoneme conversion system was developed using SMT technologies and tools and finally evaluated and compared with other works on Romanian. The existing pronunciation dictionary was extended for the domain-specific task using this phonetization system.

On the short term, the domain-adaptation methods presented in this study could be improved by tuning the domain-specific SMT system and the LM interpolation weights. On the long term, we plan to further validate the adaptation methodology by applying it for other specific domains and also for other language pairs. Another interesting perspective would be the usage of the proposed methodology when domain-specific data is available in more than one high-resourced (source) languages.

Further research might also explore the possibility of using statistical machine translation to port a larger and more general corpus, from a high-resourced language to a low-resourced language. The results of such a study might also bootstrap the development of a general corpus for the languages that do not have such linguistic resources.

Concerning the phonetization system, a future study investigating the possibility of designing a hybrid system by combining our statistical approach with a rule-based approach would be very interesting. In Romanian, there are some graphemes and groups of graphemes that are always phonetized in the same way and for these cases some rules might improve the overall performance.

REFERENCES

[1] V. Berment, "Méthodes pour informatiser les langues et les groupes de langues "peu dotées"," PhD Thesis, Université Joseph Fourier, Grenoble, France, 2004.

[2] T. Schultz and K. Kirchhoff, "*Multilingual Speech Processing*," Elsevier Academic Press, 2006.

[3] V.B. Le and L. Besacier, "Automatic speech recognition for under-resourced languages: application to Vietnamese language," *Audio, Speech, and Language Processing, IEEE Transactions on*, 17(8), pp. 1471–1482, 2009.

[4] N. Abdillahi, P. Nocera, and J-F. Bonastre, "Automatic transcription of Somali language," *ICSLP 2006*, pp. 289-292, Pittsburgh, USA.

[5] T. Pellegrini, and L. Lamel, "Investigating Automatic Decomposition for ASR in Less Represented Languages," *ICSLP 2006*, pp. 285-288, Pittsburgh, USA.

[6] P. Mihajlik, T. Fegyó, Z. Tüske, and P. Ircing, "A Morpho-graphemic Approach for the Recognition of Spontaneous Speech in Agglutinative Languages – like Hungarian," *Interspeech 2007*, pp. 1497-1500, Antwerp, Belgium.

[7] S. Stüker, "Integrating Thai grapheme based acoustic models into the ML-mix framework - for language independent and cross-language ASR," *SLTU 2008*, Hanoi, Vietnam.

[8] S. Gizaw, "Multiple pronunciation model for amharic speech recognition," *SLTU 2008*, Hanoi, Vietnam.

[9] V.B. Le, B. Bigi, L. Besacier, E. Castelli, "Using the Web for fast language model construction in minority languages," *Eurospeech 2003*, pp. 3117-3120, Geneva, Switzerland.

[10] C. Draxler, "On Web-based Speech Resource Creation for Less-Resourced Languages," *Interspeech 2007*, pp. 1509-1512, Antwerp, Belgium.

[11] J. Cai, "Transcribing southern min speech corpora with a web-based language learning system," *SLTU 2008*, Hanoi, Vietnam.

[12] H. Nakajima, H. Yamamoto, T. Watanabe, "Language Model Adaptation with Additional Text Generated by Machine Translation," *COLING 2002*, vol. 2, pp. 716-722, Taipei, Taiwan.

[13] A. Jensson, "Development of a speech recognition system for Icelandic using machine translated text," *SLTU 2008*, Hanoi, Vietnam.

[14] K. Suenderman, J. Liscombe, "Localization of speech recognition in spoken dialog systems: How machine translation can make our lives," *Interspeech 2009*, pp. 1475-1478, Brighton, U.K.

[15] S. Sam, E. Castelli, L. Besacier, "Unsupervised Acoustic Model Adaptation for Multi-Origin Non Native ASR," *Interspeech 2010*, pp. 254-257, Tokyo, Japan.

[16] B. Jabaian, L. Besacier, F. Lefevre, "Combination of Stochastic Understanding and Machine Translation Systems for Language Portability of Dialog Systems," *ICASSP 2011*, pp. 5612-5616, Prague, Czech Republic.

[17] H. Cucu, L. Besacier, C. Burileanu, A. Buzo, "Enhancing Automatic Speech Recognition for Romanian by Using Machine Translated and Web-based Text Corpora," *SPECOM 2011*, pp. 81-88, Kazan, Russia.

[18] H. Cucu, L. Besacier, C. Burileanu, A. Buzo, "Investigating the Role of Machine Translated Text in ASR Domain Adaptation: Unsupervised and Semi-supervised Methods," *ASRU 2011*, pp 260-265, Hawaii, USA.

[19] H. Cucu, L. Besacier, C. Burileanu, A. Buzo, "ASR Domain Adaptation Methods for Low-Resourced Languages: Application to Romanian Language," *EUSIPCO 2012*, pp. 1648-1452, Bucharest, Romania.

[20] D.-P. Munteanu, "Contribuții la elaborarea metodelor de recunoaştere a vorbirii în limba română," PhD Thesis, Technical Military Academy, Bucharest, Romania, 2006.

[21] C.-O. Dumitru, I. Gavăt, "Progress in Speech Recognition for Romanian Language," *Advances in Robotics, Automation and Control*, pp. 472, Vienna, Austria, 2008.

[22] C.S.Petrea, A. Buzo, H. Cucu, M. Paşca, C. Burileanu, "Speech Recognition Experimental Results for Romanian Language," *ECIT 2010*, pp. 13, Iaşi, Romania.

[23] A. Kabir, M. Giurgiu, "A Romanian Corpus for Speech Perception and Automatic Speech Recognition," *The 10th International Conference on Signal Processing, Robotics and Automation*, pp. 323-327, Cambridge, UK, 2011.

[24] M. Macoveiciuc, A. Kilgarriff, "The RoWaC Corpus and Romanian Word Sketches," in D. Tufiş, C. Forăscu (Eds.), Multilinguality and Interoperability in Language Processing with Emphasis on Romanian, pp. 151-168, Romanian Academy Publishing House, Bucharest, 2010.

[25] A. Vlad, A. Mitrea, M. Mitrea, "Printed Romanian Modeling: A Corpus Linguistic Based Study With Orthography And Punctuation Marks Included," *Computational Science and It's Applications – ICCSA 2007*, Lecture Notes in Computer Science, vol. 4705, pp. 409-423, Springer Verlag, Berlin Heidelberg, 2007.

[26] C. Ungurean, D. Burileanu, V. Popescu, C. Negrescu, A. Dervis, "Automatic Diacritics Restoration for a TTS-based E-mail Reader Application," *University Politehnica of Bucharest Scientific Bulletin*, Series C, vol. 70, no. 4, pp. 3-12, Bucharest, Romania, 2008.

[27] J. Domokoş, "Contributions on Continuous Speech Recognition and Natural Language Processing," PhD Thesis, Technical University of Cluj-Napoca, Cluj-Napoca, Romania, 2009.

[28] E. Bick, A. Greavu, "A Grammatically Annotated Corpus of Romanian Business Texts," in D. Tufiş, C. Forăscu (Eds.), Multilinguality and Interoperability in Language Processing with Emphasis on Romanian, pp. 169-182, Romanian Academy Publishing House, Bucharest, 2010.

[29] E. Oancea, I. Gavăt, O. Dumitru, D. Munteanu, "Continuous Speech Recognition for Romanian Language Based on Context Dependent Modeling," *Communications 2004*, pp. 221-224, Bucharest, Romania.

[30] D. Militaru, I. Gavăt, O. Dumitru, T. Zaharia, S. Segărceanu, S., "Protologos, System for Romanian Language Automatic Speech Recognition and Understanding," *SpeD 2009*, pp. 21 – 32, Constanța, Romania.

[31] H. Cucu, A. Buzo, C. Burileanu, "Optimization methods for large vocabulary, isolated words recognition in Romanian language," *University Politehnica of Bucharest Scientific Bulletin*, Series C, no. 2, pp. 179-192, Bucharest, Romania, 2011.

[32] L. Liță, A. Ittycheriah, S. Roukos, N. Kambhatla, "tRuEcasIng," *ACL 2003*, pp.152-159, Sapporo, Japan.

[33] M. Bisani, H. Ney, "Joint-sequence models for grapheme-to-phoneme conversion," *Speech Communications*, vol. 50, no. 5, pp. 434–451, 2008.

[34] P. Bonaventura, F. Giuliani, J.M. Garrido, I. Ortin, "Grapheme-to-phoneme transcription rules for Spanish, with application to automatic speech recognition and synthesis," *ACL 1998*, Montreal, Canada.

[35] Ş.-A. Toma, D.-P. Munteanu, "Rule-Based Automatic Phonetic Transcription for the Romanian Language", *COMPUTATIONWORLD 2009*, pp. 682-686, Athens, Greece.

[36] A. Laurent, P. Deléglise, S. Meignier, "Grapheme to phoneme conversion using an SMT system," *Interspeech 2009*, pp. 708-711, Brighton, U.K.

[37] P. Karanasou, L. Lamel, “Comparing SMT Methods for Automatic Generation of Pronunciation Variants,” *IceTAL 2010*, pp. 167, Reykjavik, Iceland.

[38] J. Billa, M. Noamany, A. Srivastava, D. Liu, R. Stone, J. Xu, J., et al., “Audio indexing of Arabic broadcast news,” *ICASSP 2002*, pp. 5-8, Orlando, USA.

[39] M. Bisani, H. Ney, “Multigram-based grapheme-to-phoneme conversion for LVCSR,” *Eurospeech 2003*, pp. 933-936, Geneva, Switzerland.

[40] P. Koehn et al., “Moses: Open Source Toolkit for Statistical Machine Translation,” *ACL 2007*, Prague, Czech Republic, 2007.

[41] M. Simard, C. Goutte, and P. Isabelle., “Statistical phrase-based post-editing,” *NAACL-HLT 2007*, pp. 508–515, Rochester, USA.

[42] H. Bonneau-Maynard, S. Rosset, C. Ayache, A. Kuhn, and D. Mostefa, “Semantic annotation of the french media dialog corpus,” *Interspeech 2005*, pp. 3457-3460, Lisbon, Portugal.

[43] P. Koehn “Europarl: A Parallel Corpus for Statistical Machine Translation,” *MT Summit 2005*, pp. 79-86, Phuket, Thailand.

[44] D. Tufiş, A. Chițu, “Automatic Insertion of Diacritics in Romanian Texts,” *International Workshop on Computational Lexicography 1999*, pp. 185-194, Pecs, Hungary.

[45] D. Tufiş, D. Ceauşu. “DIAC+: A Professional Diacritics Recovering System,” *LREC 2008*, Marrakech, Morocco.

[46] C. Ungurean, D. Burileanu, “An advanced NLP framework for high-quality Text-to-Speech synthesis,” *SpeD 2011*, pp. 1-6, Braşov, Romania.

[47] K. Papineni, S. Roukos, T. Ward, W.J. Zhu, “BLEU: a method for automatic evaluation of machine translation,” *ACL 2002*, pp. 311-318, Philadelphia, USA.

[48] N. Bertoldi, B. Haddow, J.-B. Fouet, “Improved Minimum Error Rate Training in Moses,” *The Prague Bulletin of Mathematical Linguistics*, pp. 1-11, February 2009.

[49] D. Burileanu, M. Sima, A. Neagu, “A phonetic converter for speech synthesis in Romanian,” *ICPhS 1999*, vol. 1, pp. 503-506, San Francisco, USA.

[50] D. Jitcă, H.N. Teodorescu, V. Apopei, F. Grigoraş, “An ANN-based method to improve the phonetic transcription and prosody modules of a TTS system for the Romanian language,” *SpeD 2003*, pp. 43-50, Bucharest, Romania.

[51] M.A. Ordean, A. Saupe, M. Ordean, M. Duma, G.C. Silaghi, “Enhanced rule-based phonetic transcription for the Romanian language,” *SYNASC 2009*, pp. 401-406, Timişoara, Romania.

[52] J. Domokoş, “Automated Grapheme-to-Phoneme Conversion System for Romanian”, *SpeD 2011*, pp. 1-6, Braşov, Romania.

[53] H. Cucu, "Towards a speaker-independent, large-vocabulary continuous speech recognition system for Romanian," PhD Thesis, Politehnica University of Bucharest, Romania, 2011.