ISBN 978-83-60810-14-9
ISSN 1896-7094

N-gram language models for Polish language. Basic concepts and applications in automatic speech recognition systems.

Bartosz Rapp
Laboratory of Language and Speech Technology
ul. Rubież 46, 61-612 Poznań
email: bartosz.rapp@speechlabs.pl

July 14, 2008

***Abstract*—Usage of language models in automatic speech recognition systems usually give significant quality and certainty improvement of recognition outcomes. On the other hand, wrongly chosen or trained language models can result in serious degradation not only recognition quality but also overall performance of the system. Proper selection of language material, system parameters and representation of the model itself is important task during language models construction process.**

This paper describes basic aspects of building, evaluating and applying language models for Polish language in automatic speech recognition systems, which are intended to be used by lawyer's chambers, judiciary and law enforcements.

Language modeling is a part of project which is still early stage of development and work is ongoing so only some basic concepts and ideas are presented in this paper.

I. Introduction

GENERALLY it is known that language models (LM) can be effective support for speech recognition process and rise the recognition quality even by tens of percents. This is why elements of language modeling are included in practically all modern automatic speech recognition systems (ASR). Often collected sound material contains many different kinds of noises and artifacts and the speakers utter words in incorrect or negligent way. Automatic analyses of such speech signal on acoustic models ([6]) level can be insufficient to obtain satisfying recognition correctness and certainty. Information supplied by language models can be invaluable help in this kind of situations. Language models used in ASR systems work in similar way to this in which human brain tries to recognize speech on the higher levels of functioning. In case when not all of the heard words are understandable, we are trying automatically replace these words with others—most probable ones. This replacement is done on our knowledge basis (formerly heard or read sentences etc.). We are simply choosing words in the process of statistical and semantical deduction that are the best fit. Hearing sentence "*Ala **na** kota*" we are very certain that the correct sentence should be "*Ala **ma** kota*". Our choice is driven by semantical knowledge and the fact that the second sentence we have heard so many times in our lifetime and probably this is what author of the message was trying to say. Additionally if we know the global context of whole conversation our deduction can be much more certain. Language models can also be used in similar way to correct mistakes which have been made on acoustic models level. Recognized by sound analysis, words not always will suit to the rest of sentence i.e. grammatically or semantically (although sentence is correct according to the rules of Polish language in fact it can have no real meaning and sense i.e. "*ryba jedzie na rowerze*"). As we can see language models are used to correct mistakes and errors which appear as the result of incorrect acoustic recognition or presence of artifacts in the speech signal.

Many language modeling techniques have been developed and proposed during last years. Some of them are based on "brute-force" statistical analysis (n-gram word LMs and n-gram class LMs ([1]), factored LMs etc.) and others are more formal with heavy theoretical and linguistic background (PCFG or HPSG ([2])). In this article only n-gram language models will be discussed more widely.

N-gram language model is a set of probabilities of the word sentences $P(w_i|w_1,...,w_{i-1})$. These probabilities can be estimated using following equation:

$$P(w_1,...,w_n) = \prod_{i=1}^{n} P(w_i|w_1,...,w_{i-1})$$

Unfortunately chance that out training set will contain the same word sequence $(w_1,...,w_n)$ more than couple of times is rather low. Good solution is therefore to treat the process of word generation as a Markov process (process without memory). According to Markov assumption we accept the fact that only N previous words will have the influence on what word w_n) will be. This is what we call n-gram ([1], [3]) language model[1] Selection of the right n-gram length has tremendous impact on the usefulness of language model and depends mainly on what results are expected to be achieved. Higher values of n give better knowledge about context (*discrimination*), lower n values are much more probable to

[1] n-gram means here subsequence of n words from some word sequence. n-gram which has length of 1 are called *unigrams*, 2—*bigrams*, 3—*trigrams*, 4—*tetragrams*, etc.

978-83-60810-14-9/08/$25.00 © 2008 IEEE

appearer in the text (*reliability*). In real world application the most common n values are $1 \leq n \leq 4$. It is important to mention that for a vocabulary containing $100\ 000$ words, 4-gram LM can have even $100\ 000^4$ parameters. In fact many of them will represent word sequences that are impossible to appear in real language and can be pruned. It is expected that real language model is usually far less complex than theoretical one. Unfortunately main disadvantage of n-gram language models if fact, it can only estimate probabilities of words from its vocabulary (which is given at the beginning) and adding new word to that vocabulary results in need of rebuilding whole model.

It is needed to remark here, that this project has a research status and the result, that is language models, is only a preliminary version. Main goal of this project (developed at Laboratory of Language and Speech Technology) is to design and implement technology demonstration ASR systems—intended to be used by lawyer's chambers, judiciary and police forces. Basic specifications assume real-time or near to real-time speech recognition of spontaneously dictated speech (i.e. reports, protocols, documents, statements etc.).

II. Training material selection

Selection of the training texts and language material is one of the most important tasks in LM building process. It should be done with care and some additional factors in mind. First of all training texts need to be appropriate for the occupational category for witch ASR system is intended. Although the same language, there are some slight differences in vocabulary used by lawyers, physicians, politicians and police officers. Each of these groups use some specific therms and sentences typical for the occupation. Nowadays creation of universal language model that suits all kinds of speech is practically impossible. It is worth to mention here that LMs developed as a part of this project are designed with lawyer's chambers, judiciary and police in mind. In the training set following texts are included: many kinds of newspaper texts (for general speech), professional press notes, court protocols, whitens testimonies and statements, police reports and government acts and Parliament speeches. Currently our laboratory owns almost 4GB of different types of texts and linguistic materials. The database is still under development and in near future should reach size of nearly 60GB of data. It should be pointed that while gathering more language material chance to include in this set rarely used words increases. This is why texts which will be used to build language models should be chosen with care.

In case of Polish language which is similar i.e. to Russian ([4]) or Turkish ([5]) vocabulary size should be couple hundred thousands of words. This is only preliminary estimation and further research is needed to obtain the real effective vocabulary size needed for building high quality language model.

III. Training texts preprocessing

Raw training texts will always contain some mistakes and artifacts like non existing words, misspellings etc. They contain also many unnormalized abbreviation i.e. *mgr inż.* instead of "*magister inżynier*" (*M.Sc.Eng.—Master in Science Engineer*) and numbers in mathematical (i.e. *100* instead of "*sto*") or roman notation. It is obvious that achieving high quality language models will require training materials prepared (preprocessed) in correct way. Having access to word dictionary it is possible to do some automated orthographic and spelling check (manual checking of millions of words is impossible). This is not really necessary step, but if there are possibilities to do this in automated way this kind of spell checking can be applied. It is worth to remark that having really large corpora single and rare misspellings are not harmful during language model construction. If much effort is needed to perform this step it should be skipped. Next normalization of abbreviation is needed. This is the hardest part of preprocessing because shortenings should be expanded to their correct form as should be in sentence according to grammatical and inflection rules and not only to base form i.e. "*Dzisiaj nie ma* ***prof.*** *Jarząbka*" should be "*Dzisiaj nie ma* ***profesora*** *Jarząbka*" and not "*Dzisiaj nie ma* ***profesor*** *Jarząbka*". Generally if we have access to really big corpora it is no need to bother about normalization. Even if one million sentences will be removed from set containing couple billion of sentences this would not be a problem. However some difficulties can be encountered when dealing with small corpora. In this project such relatively small corpora is used to model language specific for lawyers and police. Text material from that set contains large number of numeric values in mathematical notation, abbreviations and shortenings i.e. "KK" instead of "kodeks karny" (penal code) and special characters like i.e. \$, §, £, €. In this case normalization is task which should be performed. Normalization is the most complicated part of text preprocessing. Classic approach used while determining the correct word form requires inflectional and semantical analysis of whole sentence. Construction of such automatic analyzer is not an easy task and is not a part of this project. Second proposed approach is to create a set of transformation rules. Using these rules it will be possible to process unnormalized texts and replace all instances of shortenings and numbers, numerals with their expanded and correct form. In this experiment two different approaches will be used to create transformation rule sets. First rules set will be developed manually by linguistic expert according to grammatical and lexical rules of Polish language. As an alternative method some attempts to discover knowledge from collected data will be made. All texts will be divided into two parts. First (test set) will only contain sentences with unnormalized numbers, numerals, shortenings and abbreviations. Second (training set) will consist of sentences containing normalized numbers and numerals together with words (expanded forms) known from abbreviation and shortenings dictionary. Then knowledge discovery algorithms will be used to process sentences from second subset of our corpora on raw and POS labeled sentences. Discovered rules will be applied to the test set. Results of normalization made with manually developed transformation rules set and those discovered from examples will be compared. It is hard to evaluate which approach will be better in practical application. At this moment no goodness index has been proposed to be used to decide which rule set gives better results. At last all words should be capitalized. This will make further text processing much easier and less ambiguous i.e. same words because of capitalized letters will not be classified as different ones: "***Pies** jest najlepszym przyjacielem człowieka*" and "*Moim najlepszym przyjacielem jest **pies** Azor*". Language material preprocessed in this way is ready to be used to train language models.

IV. Language models

Having decent training material it is possible to proceed with language model construction. At this stage it is needed to recall that the generated language model is static structure and all vocabulary modifications, such as adding new text to training set, require rebuilding whole LM.

The most simple and common form of LM is a technique applying to evaluate probabilities of n-grams sequences for single words. Words, which have not been seen in training material are marked as abstract word *unk*. N-gram models for single words are rather rarely seen in practical applications, because of their requirements for memory and processing power, which result in higher LM's answer time latencies. This kind of resource demand can be unacceptable in real world and real-time application.

One among the methods designed to deal with limitations like such mentioned above is algorithm assigning words to some abstract classes on training texts statistical analysis basis (word exchange algorithm) or using some custom classification function i.e. assigning words to their real part of speech class (POS).

In this project at the early stage of work both mentioned methods are planned to be applied. Statistical derivation of abstract equivalent classes will be made using word exchange algorithm implemented in HTK toolkit. Grammatical POS classification will be made by tagging words according to Inflectional Vocabulary for Polish developed under management of Wiesław Lubaszewski ([8] and morphological analyzer *Morfeusz SIAT* designed by Zygmunt Saloni and Marcin Wolski ([9]). It is expected to achieve better results using inflectional vocabulary. With inflectional vocabulary words can be assigned to 60 POS categories. Furthermore detailed information about word form is also available so in practical applications number of tags can be significantly expanded. In case of *Morfeusz SIAT* there is less than 20 categories. Morphological analyzer contains on the other hand some additional information which are unavailable in inflectional vocabulary. As a part of this project results of POS tagging by these to tools will be compared. Unfortunately *Morfeusz SIAT* and inflectional vocabulary have nowadays some limitations mainly because of small word list. Problem arises in case of proper names, which are commonly seen in training texts and language material.

V. Probabilities estimation

Because of limited size of language material used to train the language model, some previously unseen and unknown words, will sometimes appear at LM input. Language model is therefore only some approximation of real language. Fact that some of the mentioned word have not been included in the training material does not mean that these words are incorrect on should be considered as non-existent. Occurrences of such words cannot be estimated as a 0 probability. During language model training process some probability mass is needed to be "reserved" for the event of seeing such unknown words. One of the available techniques designed to deal with this problem is discount coefficient factor. There are several algorithms used to calculate discount coefficients form which most popular are: Good-Turing algorithm and absolute discounting used i.e. in HTK toolkit.

VI. Enhancements techniques

Often some enhancement techniques are used to improve language model quality and decrease its complexity. Most commonly used ones are: the first technique prunes language model and cuts off rare words, the other one is a method of smoothing probabilities. According to this algorithm (probability smoothing) probabilities of some rare n-word sequences are replaced by probabilities of their corresponding shorter contexts (n-1-word sequences). This technique used in parallel with discounting coefficient factor reduces complexity of language model.

VII. Language model quality estimation

Nowadays most often used metric to evaluate the language model goodness or quality is *perplexity*. This measure has its roots in information theory. In language modeling *perplexity* expresses average number of word choices for current context. Lower *perplexity* value, the better language model was build (it makes lower error). To calculate *perplexity* test text and a language model itself is needed. Unfortunately *perplexity* seems not to be the best measure for evaluating language model quality. This is very simple and basic measure witch additionally highly depends on LM application domain and can be only applied to probabilistic models. Because of its weaknesses it is recommended to support it with other quality indicators such as i.e. Out-Of-Vocabulary (OOV) rate. Some more general measures should be used. Different approaches to language model evaluation which extend *perplexity* or constitute other independent quality factors has been discussed in [11].

VIII. Summary

In this article basic aspects and concepts of language modeling for applications in real world ASR systems have been discussed. Most of the experiments are conducted using HTK toolkit which is reliable, efficient and proven framework for designing and building automatic speech recognition systems. It is expected to achieve better results during further research of algorithms, techniques and methods (such as using some semantic and context knowledge in language model engine, using fuzzy logic and perhaps some derivations of formal languages elements). It is important to realize that nowadays this is still a research project.

Language model intended to be used in real-time ASR systems need to be applicable, that is, there must be some balance between efficiency and performance. It is needed to notice that this balance is really hard to determine and to achieve. According to our preliminary expectations, language model included in ASR system should increase its recognition accuracy even by 20%. Although we realize that designing good language model for languages like Polish, Turkish, Russian or Finish is quite challenging task.

REFERENCES

[1] Steve Young, Gunnar Evermann, Mark Gales, Thomas Hain, Dan Kershaw, Xunying Liu, Gareth Moor, Julian Odell, Dave Ollason, Dan Povey, Valtcho Valtchev, Phil Woodland *The HTK Book*, 2006.

[2] Adam Przepiurkowski, Anna Kupść, Małgorzata Marciniak, Agnieszka Myckowiecka *Formaly opis języka polskiego. Teoria i implementacja*, Akadeicka Oficyna Wydawnicza EXIT, Warsaw, 2002

[3] Steve Young, Gerrit Bloothooft *Corpus-based methods in language and speech processing*, Kluwer Academic Publishers, 1997.

[4] Whittaker E. W. D., Woodland P. C. *Language modeling for Russian and English using words and classes*, Computer speech & language, 2003, vol. 17, pp. 87-104

[5] Ciloglu T., Comez M., Sahin S. *Language modeling for Turkish as an agglutinative language*, Signal Processing and Communications Applications Conference, 2004. Proceedings of the IEEE 12th, 2004, pp. 461-462

[6] Xuedong Huang, Alex Acero, Hsiao-Wuen Hon *Spoken language processing*, Prentice Hall PTR, 2001.

[7] Barbara Lewandowska-Tomaszczyk *Podstawy językoznawstwa korpusowego*, Wydawnictwo Uniwersytetu Łudzkiego, 2005.

[8] Wiesław Lubaszewski, Henryk Wróbel, Marek Gajęcki, Barbara Moskal, Alicja Orzechowska, Paweł Pietras, Piotr Pisarek, Teresa Rokicka *Słownik fleksyjny języka polskiego*, Wydawnictwa Prawnicze LexisNexis, 2001, ISBN 83-7334-055-6

[9] Marcin Wolski *System znaczników morfosyntaktycznych w korpusie IPI PAN* POLONICA XII, PL ISSN 0137-9712, 2004

[10] Ying Liu, Xiaoyan Zhu *An Efficient Approach of Language Model Applying in ASR Systems*, International Journal of Information Technology, Vol. 11, No. 7, 2005

[11] Stanley Chen, Douglas Beeferman, Ronald Rosenfeld *Evaluation metrics for language models*, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213