550, rue Sherbrooke Ouest, bureau 100
Montréal (Québec) H3A 1B9
Tél. : (514) 840-1234 ; Téléc. : (514) 840-1244
888, rue St-Jean, bureau 555
Québec (Québec) G1R 5H6
Tél. : (418) 648-8080 ; Téléc. : (418) 648-8141
http ://www.crim.ca

CRIM - Documentation/Communications

Techniques for Vocabulary Selection and Word Weighting in Language Models

Technical report

CRIM-06/05-01

Maryse Boisvert
Research agent
Speech recognition group

May 2006

Collection scientifique et technique

ISBN-13: 978-2-89522-070-1
ISBN-10: 2-89522-070-0

Table des matières

1 Introduction

One of the most important components in a speech recognition system is the language model, which gives a probability to sequences of words. This step is crucial since most of the time the acoustic signal can be transcribed in many different sequences of words that have the same phonetic representation. The language model provides a way to choose the most likely amongst candidates based on lexical statistics.

A statistical language model (LM) [1] is a probability distribution $P(w_1^n)$ over strings w_1^n that attempts to reflect how frequently they occur as a sentence. By expressing various language phenomena in terms of simple parameters in a statistical model, SLMs provide an easy way to deal with complex natural language in computer. The original (and is still the most important) application of LMs is speech recognition, but LMs also play a vital role in various other natural language applications as diverse as machine translation, part-of-speech tagging, intelligent input method and Text To Speech system.

A language model modelizes $P(w_1^n)$. Without loss of generality, we can express this probability as $\prod_i P(w_i|w_1^{i-1})$ using Bayes' rule. Some of the most successful models of the past two decades are the simple n-gram models, particularly the trigram model where only the most recent two words of the history are used to condition the probability of the next word. The Markov independence hypothesis for a trigram language model is formulated as follows :

$$P(w_i|w_1^{i-1}) = P(w_i|w_{i-1}, w_{i-2})$$

A n-gram LM is typically trained on a large corpus of text. Co-occurrence statistics are collected and resulting probability distribution is smoothed to avoid zero probabilities to unseen events. The parameters of a trigram LM can be estimated with maximum likelihood estimation (MLE). Although n-gram language models have been found to work well in practice, there are still many recognition errors due to the absence of grammatical and semantic information in them.

After having completed studies, we noticed that there were three major types of errors in the recognized sentences. The first and most important type is out-of-vocabulary (OOV) words, i.e. words that are not in the vocabulary known to the recognizer. The second type of errors occurs when a word is replaced by many smaller words that have the same phonetic transcription as the uttered word and have a larger combined probability of appearing. The last type is caused by gender and number disagreement, since most of the time, taking the plural form of a noun, an adjective or a past participle results in an homophone.

In the next sections, we provide solutions that allows us to reduce the number of errors of the different types listed above. Section 2 presents an algorithm to reduce the number of OOV words ; section 3 describes how we adapt the system so words have appropriate probabilities ; section 4 proposes a model to cope with gender and number agreement.

2 Vocabulary Selection

A speech recognizer converts the observed acoustic signal into the corresponding orthographic representation of the spoken utterance. The system chooses its candidates from a finite vocabulary of words that can be recognized. The error rate of a speech recognizer is no less than the percentage of spoken words that are not in its vocabulary, hence a major part of building a language model is to select a vocabulary that will have maximal coverage on new text spoken to the recognizer. In general, a corpus of text is used in conjunction with dictionaries to determine appropriate vocabularies. The selection of the vocabulary highly depends on the domain being modelized hence the vocabulary is different for every task.

There are many machine learning algorithms to automatically induce the vocabulary from a corpus of text. As an example, we can select as the vocabulary the N most frequent words. When many sources of text are used, there exists different schemes to assign a weight to each source. The weights can then be converted to word weights according to the frequencies of words in each of the sources. In the next paragraphs we discuss the implementation of such a weighting algorithm which is is used to select the vocabulary for specific tasks.

Typically, when selecting a vocabulary from a corpus, words need to be ranked according to a weight that defines their importance for the domain. When many texts are available, the weighted count $C(w_i)$ of word w_i is given by linear interpolation of its weighted count on each text :

$$C(w_i) = \sum_k \lambda_k C_k(w_i)$$

The parameters λ_k can be determined in many ways. A simple approach uses the size of a text as its weight in the interpolation [2]. Another approach involves mixture weights estimated on a sample of text close to the domain being modelized [3]. The weight of a text is determined automatically by linear interpolation with the Expectation-Maximization (EM) algorithm [4] on the validation set. The weights are then normalized by text sizes. A sample of text close to the target domain is used as a validation set.

Our implementation is based on the latter. We estimate a unigram language model on each of the source texts. Then the EM algorithm is used to find the interpolation weights that maximize the likelihood of the validation set given the models. We convert the model mixture weights into the text weight in proportion to their text size :

$$w_i = \frac{\lambda_i * m_0}{\lambda_0 * m_i}$$

where m_i is the size of source i and λ_i is its interpolation weight. The new vocabulary is defined by the mixed texts according to the ranking of word frequencies.

We applied this technique to select the vocabulary to be included in the language model used for subtitling NHL hockey games. The vocabulary is derived from the same corpus used to train the parameters of the LM. The training set is made of texts from sports section of La Presse and transcriptions of hockey games. To measure the performance of a given vocabulary, we compute the OOV rate, which is the number of occurrences of words not in the vocabulary on a text. The OOV rate of the vocabulary V on the text $T = w_1, \ldots, w_N$ is given by :

$$OOV(T) = \frac{\sum_{i=1}^{N}(1 - I(w_i, V))}{N}$$

where $I(w_i, V)$ is the indicator function of w_i belonging to the vocabulary V. The OOV rate is a standard measure of the coverage of the vocabulary and should be as small as possible.

We compare the OOV rate on the transcription of a hockey game of a vocabulary derived from a dictionary and a vocabulary produced by the algorithm described above. The former gets 2.02% while the latter obtains 1.22%.

3 Language Model Adaptation

A background language model is estimated from a large corpus, but the resulting parameters are in general not able to cope with significant variations in language such as topic shifts and domain changes. Therefore, as new data becomes available, we constrain the background LM to the features extracted from the adaptation data. The minimum discriminant information (MDI) [5] estimation is used to fit as close as possible the background model (in terms of the Kullback-Leibler distance) while satisfying the constraints derived from the adaptation data.

Our implementation follows [6]. A unigram LM is estimated on the adaptation data and is interpolated with the background unigram. Then, the higher order n-grams and back-off weights are updated to reflect the changes in the probability distribution over the words in the vocabulary.

One problem with MDI adapation is related to words that appear in only one of the two data sets (background data or adaptation data). In a language model, the class of all words unknown to that model (i.e not in the vocabulary) receives a probability that is estimated based on the out-of-vocabulary rate on the training set. When interpolating two LMs, the words in these unknown classes can potentially be given a zero probability, which leads to underestimation of the probability of occurrence of that word. We therefore need to provide an estimate of the probability of a word belonging to the unknown class of a LM.

Before interpolating the two unigrams, we assign a unigram probability to every word in the joint vocabulary in both the background model and adaptation model and renormalize the unigrams so they sum to one. The new unigram probability for a word in a model is given by the model

probability if the unigram exists and by the smallest possible probability if it does not. The smallest probability in a model is at least the probability given to an unseen event by the smoothing scheme applied to word counts in the LM construction.

We applied this procedure to the adaptation of a LM for NHL hockey games. Hockey players names seen in the background training set get on average a probability of 10^{-06}. As an example, "Alexander Perezhogin"[1] was seen in transcriptions of hockey games (used as part of the training corpus) and obtains a probability of $2.133 * 10^{-05}$ in the background LM. The probability of an unknown word is 10^{-02} in the same model, which is too high to be assigned to unseen words. The names of two hockey players, "Brian Ekland" and "Nick Tarnasky", were included in the adapted LM by the procedure described above. They both obtain $5.525 * 10^{-06}$ as a unigram probability in the adapted language model, which is close to the average probability in the background LM, and is a reasonable estimation of the probability of an unseen hockey player.

This variation in the MDI algorithm allows us to specify an arbitrary vocabulary to the adaptation procedure and obtain estimates of the probability of any word appearing in a text, just as if new words were included in the vocabulary when training the original language model.

4 Gender and Number Agreement

The language model alone is inadequate to realize gender and number agreement because the only information one can rely on is at the lexical level. Since many of the inflected forms have the same pronunciation, it is often critical to choose the right one. In many cases, the most frequent from will be chosen. A simple approach to realize the gender and number agreement is to incorporate directly in the language model the grammatical information. This results in an explosion of the number of LM parameters that need to be estimated. This method cannot be implemented in practice because there is not enough data to provide a reliable estimate of the LM parameters.

We propose a post-processing algorithm to rescore a lattice of homophones created with the output sentence of the recognizer. The idea is to build an external model to the LM that uses grammatical information to give a probability to words according to their flexion. A class-based language model is used to implement this idea, where the classes are the parts-of-speech. The parts-of-speech (POS) of a word are its possible grammatical categories. This LM allows us to give a probability to a sequence of POS tags. For example, it is more probable that a feminine adjective is followed by a feminine noun than it is followed by a masculine one in a sentence. To implement this idea, we need a dictionary listing all the possible parts-of-speech for each word and an algorithm that gives the parts-of-speech of a word in its context. We present in Section4.2 a probabilistic model which relies on an algorithm for tagging words with their parts-of-speech, and which is based on the use of electronic dictionaries. First, we introduce the electronic data-

[1]Note that players' names appear as expressions in the corpus and are therefore considered as single words.

TAB. 1: Unitex grammatical codes for French and their corresponding grammatical category.

Code	Category
A	Adjective
ADV	Adverb
CONJC	Conjonction of coordination
CONJS	Conjonction of subordination
DET	Determinant
INTJ	Interjection
N	Noun
PREP	Preposition
PREPDET	Preposition determinant
PRO	Pronoun
V	Verb

TAB. 2. Unitex flexion codes for gender and number.

Code	Category
m	Masculine
f	Feminine
s	Singular
p	Plural

base that was used to retrieve the POS.

4.1 Unitex

The possible POS for a word were obtained with Unitex, a corpus processing system based on automata. Tags are retrieved via electronic dictionaries in the DELAF format [2]. Words in the dictionary are tagged with lexical, semantic and morphological information. Since keeping all types of information would result in too many tags, we focus on the lexical and morphological parts of the tag for a given word. For example, the two possible tags of the French word “pense” are “N :fs” and “V :Kfs”, respectively referring to the feminine singular noun and the feminine singular past participle of the verb “penser”. The most common grammatical and flexion tags for French are listed in Tables 1, 2 and 3.

[2] See http ://www-igm.univ-mlv.fr/∼unitex/ for more information.

TAB. 3: Unitex flexion codes for French verbs and their corresponding verb tense. Tenses are listed in French.

Code	Category
P	Indicatif présent
I	Indicatif imparfait
S	Subjonctif présent
T	Subjonctif imparfait
Y	Impratif présent
C	Conditionnel présent
J	Passé simple
W	Infinitif
G	Participe présent
K	Participe passé
F	Futur

4.2 Part-of-Speech Tagging

Part-of-speech tagging [7] is the process of marking up the words in a text with their corresponding parts-of-speech. There are many approaches to automated POS tagging. An important distinction that can be made among POS taggers is based on the degree of automation of the training process. The terms commonly applied to this distinction are supervised vs. unsupervised. Supervised taggers typically rely on pre-tagged corpora to serve as the basis for creating any tools to be used throughout the tagging process, while unsupervised models are those which do not require a tagged corpus. In particular, unsupervised stochastic taggers use sophisticated computational methods to automatically induce the probabilistic information on words and tags.

One of the dominant stochastic approaches is based on Hidden Markov Models [8]. A Hidden Markov Model (HMM) [9] specifies a joint probability distribution over a word and tag sequence. Each word is assumed conditionally independent of the remaining words and tags given its part-of-speech. Tags are assumed conditionally independent of remaining tags given the previous tag. The parameters of a HMM are given by transition probabilities on the sequence of tags $P(t_i|t_{i-1})$ and by emission probabilities of a word given a tag $P(w_i|t_i)$. These parameters are estimated on an untagged corpus with the Expectation-Maximisation (EM) algorithm.

The parameters of the HMM described above define a class-based language model where

$$P(w_i|w_1^{i-1}) = P(w_i|t_i) * P(t_i|t_{i-1})$$

The transition probabilities define a bigram on the tag set. The resulting language model has fewer parameters since the tag set is much smaller than the vocabulary.

The POS bigram $P(t_i|t_{i-1})$, emission probabilities $P(w_i|t_i)$ and training set (untagged sequence of words) are all implemented as finite-state transducers. An approximation of the EM algorithm is used to train the model parameters, where we repeatedly update the parameters by counting events (E-step) in the path with the highest probability (M-step) in the composition of the training set, emission model and POS bigram. The model parameters are initialized uniformly, i.e. for each tag $t \in T$

$$P(t'|t) = \frac{1}{|T|} P(w|t) = \frac{1}{\sum_{w \in T_t}}$$

where T_t corresponds to the set of words having t as a possible POS tag. The parameters are updated by counting, normalizing and smoothing the frequencies on the most probable path in the composed transducer.

The algorithm described above was trained on a portion of the archives of the newspaper *La Presse* for the year 2003. The training set includes over 2 millions of words and contains 77645 different word forms. The set of all possible tags for these words is of size 115. The validation set is composed with over 200000 words taken from the same archive. Figure 1 shows the bestpath of the test sentence "il est gentil". We see that the adjective "gentil" is maculine singular which agrees with the pronoun "il". The verb "est" is marked as the verb "tre" conjugated at the third person singular, at the simple present tense, which is the correct part-of-speech[3]. This means that the model gives a better probability to the verb when it is preceded by a pronoun than to the noun. The weight (cost) associated with each arc is indicated on the arcs.

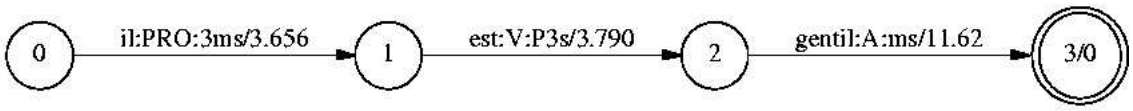

FIG. 1. Bestpath of a test sentence.

5 Conclusion and Future Work

In this report, we described how we handled errors related to vocabulary in our speech recognition setup. The three sources of errors we focused on were out-of-vocabulary words, homophones and gender and number agreement.

[3]The word "est" is also a noun.

To minimize the first type of errors, we use a vocabulary selection algorithm that optimizes the coverage of the vocabulary on a sample target of text. In addition, we propose a language model adaptation scheme that allows us to include new words in the vocabulary and provides an estimate of their probability of appearance. This technique is also used to adress the second type of errors, which are errors caused by homophones. The adaptation algorithm adjust the weights of the exisiting n-grams so that the model fits more adequately the domain as new data become available. We noticed that this technique helped to give a higher probability to words composed of many syllables that were usually replaced in the recognition by many smaller words that have the same phonetics.

We are still working on a solution to gender and number agreement. So far, we have constructed a class-based language model for the parts-of-speech of words. The language model gives a better probability to sequences of words that agree with each other on gender and number. We project to use this model to rescore the speech recognizer hypothesis (output). We think that might allow us to actually include words that are outside the vocabulary scope in the output sentence. As an example, we might not need to keep all flexions of a word in the vocabulary. The right flexion of a word would be chosen based on the most probable sequence of parts-of-speech in the lexical context.

Références

[1] C. Manning et H. Schütze, *Foundations of Statistical Natural Language Processing*, MIT Press, Cambridge, 1999.

[2] Anand Venkataraman et Wen Wang, "Techniques for effective vocabulary selection," in *Proc. of Eurospeech'2003*, Geneva, Switzerland, 2003, pp. 245–248.

[3] T. Imai A. Kobayashi, K. Onoe et A. Ando, "Time dependent language model for broadcast news transcription and its post-correction," in *Proc. Int. Conf. on Spoken Language Processing*, Sydney, Australia, 1998, pp. 2435–2438.

[4] A. P. Dempster, N. M. Laird, et D. B. Rubin, "Maximum-likelihood from incomplete data via the EM algorithm," *Journal of Royal Statistical Society B*, vol. 39, pp. 1–38, 1977.

[5] S. Della Pietra, V. Della Pietra, R. L. Mercer, et S. Roukos, "Adaptive language modeling using minimum discriminant estimation," in *HLT '91 : Proceedings of the workshop on Speech and Natural Language*, Morristown, NJ, USA, 1992, pp. 103–106, Association for Computational Linguistics.

[6] M. Federico, "Efficient language model adaptation through mdi estimation," in *Proc. of Eurospeech'99*, Budapest, Hungary, 1999, pp. 1583–1586.

[7] Eugene Charniak, "Statistical techniques for natural language parsing," *AI Magazine*, vol. 18, no. 4, pp. 33–44, 1997.

[8] Qin Iris Wang et Dale Schuurmans, "Improved estimation for unsupervised part-of-speech tagging," in *Proceedings of the 2005 IEEE International Conference on Natural Language*

Processing and Knowledge Engineering (IEEE NLP-KE'2005), Wuhan, China, 2005, pp. 219–224.

[9] Lawrence R. Rabiner, "A tutorial on Hidden Markov Models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.