Fast call-classification system development without in-domain training data

Christophe Servan, Frederic Bechet

LIA - University of Avignon, BP1228 84911 Avignon cedex 09 France
{*christophe.servan,frederic.bechet*}@*univ-avignon.fr*

Abstract

This paper presents a new method for the fast development of call-routing systems based on pre-existing corpora and knowledge databases. This method pushes forward the reduction of specific data collection and annotation for developing a new call-classification system. No specific data collection is needed for training both for the Automatic Speech Recognition (ASR) and classification models. The main idea is to re-use existing data to train the models, according to *a priori* knowledge on the task targeted. The experimental framework used in this study is a call-routing system applied to a civil service information telephone application. All the *a priori* knowledge used to develop the system is extracted from the civil service information website as well as pre-existing corpora. The evaluation of our strategy has been made on a test corpus containing 216 utterances recorded by 10 different speakers.

1. Introduction

Call-routing is a task which consists in giving a label (or *call-type*) to a spoken utterance, each label corresponding to a given service in a telephone application. This can be considered as the first level in a Spoken Language Understanding (SLU) process: the call-types corresponds to the general thematic of an utterance, for example the kind of request in a Spoken Dialog System (SDS) like *How May I Help You?* [5]. This task has been well studied and numerous strategies have been proposed, most of them representing this problem as a classification task, using state-of-the-art classifiers like Support Vector Machines [6] or Boosting algorithms [10].

The major drawback of the techniques proposed is the need for an annotated training corpus on which the classification methods can be trained. Examples of users' requests with their corresponding call-types have to be collected, for example with a Wizard-Of-Oz method. This is a major constraint as such a collect is costly and sometime difficult to set up. The manual annotation process of the collected data is also very costly and is one of the major burdens in the deployment of an automated service. Several studies have proposed to alleviate this need for manual annotation or even suppress it, at least for the manual word transcriptions of the training utterances. For example by performing a direct matching between utterances and call-types at the phonetic or morpheme levels [1, 7]. However all these studies still need a training corpus in order to learn the projection from the spoken utterances to the call-type labels.

The study presented in this paper push forward the reduction of specific data collection and annotation for developing a new call-classification system. We present a method for developing a first prototype with no collect at all of training corpus, both for the Automatic Speech Recognition (ASR) and classification models. The main idea is to re-use existing data to train the models, according to *a priori* knowledge on the task targeted.

The experimental framework used in this study is a call-routing system applied to a civil service information telephone application. All the *a priori* knowledge used to develop the system is extracted from the civil service information website as well as pre-existing corpora. The evaluation of our strategy has been made on a test corpus containing 216 utterances recorded by 10 different speakers.

family/health/work	insurance	elections
environment	foreigner	taxes
ID papers	disabled person	retirement
transport	holidays	non-profit organisation

Table 1: Thematic used as call-type labels in the call-classification system

2. Experimental framework

2.1. Task description

The task targeted is a civil service information call-routing application. Each user calling the application is automatically routed to the service in charge of the request expressed. These requests are administrative questions about schools, driving licence, taxes, paper IDs, ... A set of 12 thematic, considered as call-type labels in this study have been defined. They are listed in table 1. This is an example of questions that can be addressed to the system:

```
I want to renew my driving licence
```

2.2. A priori knowledge on the task

The only in-domain data available to us on the task is the civil service information WEB site. Following the approach proposed in [3], we consider a WEB site as a structured database of linguistic information from which the resources needed to build an automatic speech processing application can be derived. Three kinds of linguistic resources are needed in order to build a call-classification system: a lexicon; a text corpus on which the ASR language model can be trained; a set of pairs *sentence/call-type label* on which the classification model can be learnt.

All the text data contained in the civil service information WEB site[1] has been gathered. Thanks to the structure of the

This work is supported by the 6th Framework Research Programme of the European Union (EU), Project LUNA, IST contract no 33549. The authors would like to thank the EU for the financial support. For more information about the LUNA project, please visit the project home-page, www.ist-luna.eu.

[1] http://www.vaucluse.fr/

WEB site we have label each portion of text collected with one or more thematic labels, among our 12 call-type label set. A first lexicon and a text corpus containing all the thematic segments extracted constitute our only in-domain data.

To validate the assumption that the data collected was relevant to our call-routing application we performed the following experiments:

- A corpus of pairs sentence/call-type label has been extracted from the WEB data, more specifically from the **Frequently Asked Questions** (FAQ) section. This corpus, called *C_FAQ* in this paper, contains 1980 pairs.
- Following a *leave-one-out* scheme, we trained a text classifier on every partitions of 1979 examples of *C_FAQ* and tested the model on the remaining example. We used the text classifier *BoosTexter* based on the AdaBoost algorithm [8].
- After classifying the whole *C_FAQ* corpus we obtained on correct classification rate of 80% on the set of 12 call-type labels.

This classification rate was considered as acceptable with respect to the small size of the training corpus and the corpus *C_FAQ* was chosen to be our training corpus for the call-type classification models.

2.3. Evaluation corpus

The goal of this study is to develop a call-routing system without any data collection process for training the different ASR and SLU models. However we need to perform such a data collection in order to evaluate the system developed. Therefore we have recorded a test corpus, called *C_TEST*, containing 216 messages expressed by 10 speakers (male and female). The only constraint given to the speakers was to express one or several requests about each topic corresponding to our call-type labels presented in table 1.

After an *a posteriori* analysis of the *C_TEST* corpus, we categorized the messages according to three levels of complexity:

- The first one corresponds to the *easy* messages; these are short messages expressing a direct request in the topic considered. For example: "*I have a question about my retirement plan*". A set of 77 messages are considered *easy*, they are grouped in the corpus *C_TEST_easy*.
- The second level represents the *moderate* messages; these are longer messages containing some comments in addition to a direct request. For example: "*I just bought a new car and I would like to get a registration number*". There are 94 messages labelled with this complexity levels, grouped into the *C_TEST_moderate* corpus.
- The last one contains the *difficult* messages. These are long messages (twice as long as the previous ones), containing a lot of comments and out-of-domain utterances, without necessarily a clear request. For example: "*yes I just moved from Paris to Avignon and I was told well when I've asked the previous operator I was in communication to ...* ". They are 45 difficult messages, grouped into the *C_TEST_difficult* corpus.

All the evaluation is going to be made according to these levels of difficulties.

LM	WEB	EPAC	RITEL	ALL
# words	16k	114k	46k	176k
Perplexity	198	369	345	96

Table 2: Perplexity measures on the corpus *C_TEST* according to the LM used.

3. Automatic Speech Recognition with no specific data collection

3.1. Building a Language Model from pre-existing corpora

We used 3 different corpora to build our 3-gram Language Model (LM). The first one is the text corpus gathered on the WEB and presented in the previous section (including the *C_FAQ* corpus). It contains in-domain words related to the call-classification task but is not adequate to model spontaneous speech. The second corpus contains transcriptions of spontaneous conversational speech collected through the French project *EPAC*. This corpus is clearly out-of-domain however it contains a lot of spontaneous speech expressions that can be useful in the application targeted. The last corpus comes from the *RITEL* project, which aims to develop a spoken open-domain Question&Answer system [4]. This corpus is interesting because it contains many expressions of requests.

The lexicon used to train our LM is made of 2872 words extracted from these 3 different corpora. The Out-Of-Vocabulary rate of this lexicon on the *C_TEST* corpus is 6.7%. The size of each corpora and the perplexity measure obtained on the *C_TEST* corpus with a 3-gram LM trained on each of them is presented in table 2. As we can see the lowest perplexity is obtained with the *WEB* corpus, which is normal since it contain in-domain data. However by adding to this corpus the data coming from the *EPAC* corpus (for the spontaneous speech effects) and the *RITEL* corpus (for the request expressions), we obtain a very significant reduction of perplexity with a value of 96. The merging process of these three LMs is straightforward since we didn't have any adaptation corpus: they all have the same weight in the combined LM.

For the sake of comparison if we only merge the corpora *WEB* and *EPAC* we obtain a perplexity of 126 and if we merge *WEB* and *RITEL* only we obtain a perplexity measure of 105.

3.2. ASR output

The ASR module used in this study is the SPEERAL ASR system developed at the University of Avignon. The acoustic models used have been trained on the ESTER corpus and not adapted to this specific task. The ASR lexicon of 2872 words has been phonetically transcribed by means of the grapheme-to-phoneme transcription tool LIA_PHON[2].

The output of the SPEERAL system for each spoken message is a word lattice.

The ASR performance obtained on the *C_TEST* corpus is presented in table 3. As we can see the average Word Error Rate (WER) is 54.2%. This is a high WER but we have to take into consideration that we didn't use any adaptation corpus for tuning our ASR models. Indeed this WER is highly dependent on the complexity of the messages: for the *easy* messages the WER is 28.1 and it reaches 73.5 for the *difficult* messages mainly because of their out-of-domain content outlined by the big OOV rate on this particular corpus (15%).

[2] LIA_PHON *http://www.lia.univ-avignon.fr/chercheurs/bechet/*

C_TEST	*easy*	*moderate*	*difficult*	*all*
size	77	94	45	216
# words	808	1209	914	2931
OOV rate	0.5%	10.8%	15%	6.7%
WER	28.1	52.7	73.5	54.2

Table 3: Word error rates obtained on the different test corpora

4. Spoken Language Understanding with no specific data collection

As presented in the introduction, for a call-routing task, SLU consists in classifying a spoken utterance into a set of predefined classes. This can be done directly from the words hypothesized by the ASR module or by using an intermediate step that first translates a word sequence into a sequence of basic semantic units often called *concepts*. This is the approach used in the European project LUNA[3] where the first step of any SLU system is this word-to-concept translation process.

To obtain such concepts for our call-classification task, without collecting any data, we used the following approach:

- Following the same approach as the one used for obtaining a Language Model from various corpora, we used knowledge resources coming from other applications in order to build our concept ontology. Thanks to our contribution to the EVALDA/MEDIA evaluation program [2], we used the most generic part of the MEDIA ontology (time and date expressions, amounts, quantifiers, ...) to start our concept list.
- Then we extracted some domain-specific concepts related to the topics of our application from the *WEB* corpus. This was done by looking for keywords in the civil service information WEB site. We decided to keep every words between the HTML markers Bold <B>, Italic <I>, and links <A>. This selection gave us about 4500 different keywords. In order to sort them and group them into concepts, we used the *Term Frequency - Inverse Document Frequency* (TF.IDF) measure combined with a confusion matrix. Extracting the most specific keywords, we obtained 20 concepts to be added to our generic ontology.

In total we had a set of 44 concepts, each of them described either by regular grammars (for example for the date or amount concepts coming from the MEDIA application) or by a set of possible surface form found in the *WEB* corpus for the specific concepts.

These concepts have been added as features in our call-type classification process. This has been done by adding the translations from word to concept of the *C_FAQ* corpus to the classifier training corpus. The same leave-one-out validation experiment as the one presented in section 2 has been performed with the concept information, leading to a correct classification rate of 85%, an absolute 5% improvement compared to the words alone.

For classifying the spoken messages of the *C_TEST* corpus we compare 3 different approaches:

1. *baseline*: this approach consists in classifying the 1-best word string produced by the ASR module, the only features used are the word recognized;

generic concepts			
localization	address	urban district	zip code
state	country	neighbourhood	area
street	city	number	percent
name	amount	answer	request
time-year	time-age	time-date	time-hour
time-day	time-month		
specific concepts			
service	army	insurance	association
bank	citizenship	school	Europe
family	nature	training	handicap
taxes	justice	accommodation	medical
papers	work	activities	vehicles

Table 4: Concept list obtained from *MEDIA* and the *WEB* corpus

2. *word/concept sequence*: following the integrated ASR/SLU approach proposed in [9], we look at the same time for the best sequence of word and concepts in the word lattice produced by the ASR module;
3. *bag of word/concept*: because of the poor word transcription results obtained with the ASR models, we evaluated a new method that consists in obtaining directly a bag of concept with their word support from the ASR word lattice.

For the last two approaches, the features used in the classification process are both the words and the concepts.

The new *bag of word/concept* approach proposed in this paper is described in the next section.

5. The *bag of word/concept* approach

In the SLU strategy developed at the University of Avignon [9], interpretation starts with a translation process in which stochastic Language Models are implemented by Finite State Machines (FSM) which output labels for semantic constituents. These semantic constituents are called *concept tags* and are noted C_i. They correspond to the 44 concept tags defined in the ontology. To each concept tag C_i is attached the word string supporting the concept. There is an FSM for each elementary conceptual constituent. Each FSM implements a finite state approximation of a natural language grammar. These FSMs are transducers that take words at the input and output the concept tag conveyed by the accepted phrase. All these transducers are grouped together into a single transducer, called *Concept FSM*, which is the union of all of them. During the decoding of a message, a first ASR module generates a word graph (G_W) which is composed with the transducers *Concept FSM*; the result of this composition is the transducer T_{WC} (a path in T_{WC} is either a word string if one keeps only the input symbols or a concept tag string if one considers the output symbols of the transducer) like in figure 1.

The *bag of word/concept* approach consists in filtering this transducer T_{WC} with n filters corresponding to the n concepts of our ontology (44 in these experiments). Each filter F_i is a simple FSM acceptor that accepts all the path going through at least one concept C_i. The best path $\widehat{WC}_i$ of each intersection between T_{WC} and F_i is kept. Of course this intersection can be empty, in this case the best path is also empty. The probability $P(C_i|A)$ (with A being the speech signal) is obtained with the

[3] http://www.ist-luna.eu/

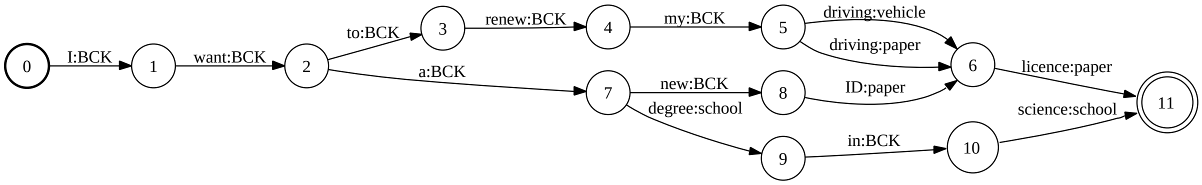

Figure 1: Example of lattice of words/concept T_{WC}

C_TEST	*ref W*	*ref W/C*	*ASR W*	*ASR W/C*	*ASR BoW/C*
easy	79.2	79.2	70.1	74.0	80.5
mod.	74.5	80.8	65.9	69.1	71.2
diff.	51.1	73.3	40.0	55.5	57.7
all	71.3	78.7	62.1	68.1	71.7

Table 5: Classification score according to the SLU method used and the complexity of the test corpus

following formula:

$$P(C_i|A) = \frac{P(\widehat{WC_i}|A)}{P(\hat{W}|A)} \qquad (1)$$

with $\hat{W}$ being the 1-best word hypothesis of the word lattice. The probabilities $P(\widehat{WC_i}|A)$ and $P(\hat{W}|A)$ are given only by the ASR acoustic and language models.

After this filtering process we obtain 44 probabilities $P(C_i|A)$ for each concept C_i with the best word string support found in the ASR lattice. If the filtering process produced an empty intersection for a concept k, then the probability $P(C_k|A)$ is set to 0.

For example, after filtering the transducer presented in figure 1, we obtain the following bag of word/concept:
concept=paper, *words=(driving licence, licence, ID)*
concept=vehicle, *words=(driving)*
concept=school, *words=(degree, science)*

This bag can be filtered according to the probabilities $P(C_i|A)$, for example the concept *school* can be discarded if the probability is too low $P(school|A)$ is too low in T_{WC}.

6. Experiments

Table 5 shows the results obtained by the 3 different classification methods on the corpus *C_TEST*. Column *ref W* indicates the performance obtained on the manual transcriptions of the text corpus (WER=0%); *ref W/C* is the best sequence of word/concept obtained on the manual transcriptions; *ASR W* is the classification method only on words; *ASR W/C* reports the results obtained with the best sequence of word/concept; *ASR BoW/C* is the new method of bag of word/concept presented here. As we can see using concepts in addition to words gives a clear improvement toward the use of words only. The bag of word/concept method adds more robustness to the classification process by being less dependent of bad transcriptions produced by poorly trained ASR models.

7. Conclusion

We have shown in this study how a call-classification process can be integrated into the ASR and SLU processes. The conceptual classification proposed is trained with very few data and is directly integrated into the ASR process. This approach allows us to keep the probabilistic search space on sequences of words produced by the ASR module and to project it to a probabilistic search space of sequences of concepts. Theses sequences once filtered, can be exploited using a classifier, thanks to a bag of word and concept strategy. Even with high values of WER, the call-classification performance obtained on the test corpus with our method is acceptable.

8. References

[1] H. Alshawi. Effective utterance classification with unsupervised phonotactic models. *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 1–7, 2003.

[2] Hélene Bonneau-Maynard, Christelle Ayache, Fréd'eric Béchet, Alexandre Denis, Anne Khun, Fabrice Lefevre, Djamel Mostefa, Mathieu Quinard, Christophe Servan, and Jeanne Villaneau. Results of french evalda-media evaluation campaign for litteral understanding. pages 2054–2059, 2006.

[3] J. Feng, S. Bangalore, and M. Rahim. WebTalk: mining Websites for automatically building dialog systems. *Automatic Speech Recognition and Understanding, 2003. ASRU'03. 2003 IEEE Workshop on*, pages 168–173.

[4] Olivier Galibert, Gabreil Illouz, and Sophie Rosset. Ritel: An open-domain, human-computer dialog system. In *INTERSPEECH*, 2006.

[5] A. L. Gorin, G. Riccardi, and J.H. Wright. How May I Help You ? In *Speech Communication*, volume 23, pages 113–127, 1997.

[6] P. Haffner, G. Tur, and JH Wright. Optimizing SVMs for complex call classification. *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, 1, 2003.

[7] Q. Huang and S.J. Cox. Automatic Call-Routing Without Transcriptions. *Eighth European Conference on Speech Communication and Technology*, 2003.

[8] Robert E. Schapire and Yoram Singer. BoosTexter: A boosting-based system for text categorization. *Machine Learning*, 39:135–168, 2000.

[9] Christophe Servan, Christian Raymond, Frederic Bechet, and Pascal Nocera. Conceptual decoding from word lattices: application to the spoken coprus media. In *INTERSPEECH - ICSLP*, page 4, September 2006.

[10] I. Zitouni et al. Boosting and combination of classifiers for natural language call routing systems. *Speech Communication*, 41(4):647–661, 2003.