Resume Information Extraction with Cascaded Hybrid Model

Kun Yu
Department of Computer Science and Technology
University of Science and Technology of China
Hefei, Anhui, China, 230027
yukun@mail.ustc.edu.cn

Gang Guan
Department of Electronic Engineering
Tsinghua University
Bejing, China, 100084
guangang@tsinghua.org.cn

Ming Zhou
Microsoft Research Asia
5F Sigma Center, No.49 Zhichun Road, Haidian
Bejing, China, 100080
mingzhou@microsoft.com

Abstract

This paper presents an effective approach for resume information extraction to support automatic resume management and routing. A cascaded information extraction (IE) framework is designed. In the first pass, a resume is segmented into a consecutive blocks attached with labels indicating the information types. Then in the second pass, the detailed information, such as *Name* and *Address*, are identified in certain blocks (e.g. blocks labelled with *Personal Information*), instead of searching globally in the entire resume. The most appropriate model is selected through experiments for each IE task in different passes. The experimental results show that this cascaded hybrid model achieves better F-score than flat models that do not apply the hierarchical structure of resumes. It also shows that applying different IE models in different passes according to the contextual structure is effective.

1 Introduction

Big enterprises and head-hunters receive hundreds of resumes from job applicants every day. Automatically extracting structured information from resumes of different styles and formats is needed to support the automatic construction of database, searching and resume routing. The definition of resume information fields varies in different applications. Normally, resume information is described as a hierarchical structure with two layers. The first layer is composed of consecutive general information blocks such as *Personal Information*, *Education* etc. Then within each general information block, detailed information pieces can be found, e.g., in *Personal Information* block, detailed information such as *Name*, *Address*, *Email* etc. can be further extracted.

Info Hierarchy		Info Type (Label)
General Info		Personal Information(G_1); Education(G_2); Research Experience(G_3); Award(G_4); Activity(G_5); Interests(G_6); Skill(G_7)
Detailed Info	Personal Detailed Info (*Personal Information*)	Name(P_1); Gender(P_2); Birthday(P_3); Address(P_4); Zip code(P_5); Phone(P_6); Mobile(P_7); Email(P_8); Registered Residence(P_9); Marriage(P_{10}); Residence(P_{11}); Graduation School(P_{12}); Degree(P_{13}); Major(P_{14})
	Educational Detailed Info (*Education*)	Graduation School(D_1); Degree(D_2); Major(D_3); Department(D_4)

Table 1. Predefined information types.

Based on the requirements of an ongoing recruitment management system which incorporates database construction with IE technologies and resume recommendation (routing), as shown in Table 1, 7 general information fields are defined. Then, for *Personal Information*, 14 detailed information fields are designed; for *Education*, 4 detailed information fields are designed. The IE task, as exemplified in Figure 1, includes segmenting a resume into consecutive blocks labelled with general information types, and further extracting the detailed information such as *Name* and *Address* from certain blocks.

Extracting information from resumes with high precision and recall is not an easy task. In spite of

The research was carried out in Microsoft Research Asia.

Proceedings of the 43rd Annual Meeting of the ACL, pages 499–506,
Ann Arbor, June 2005.

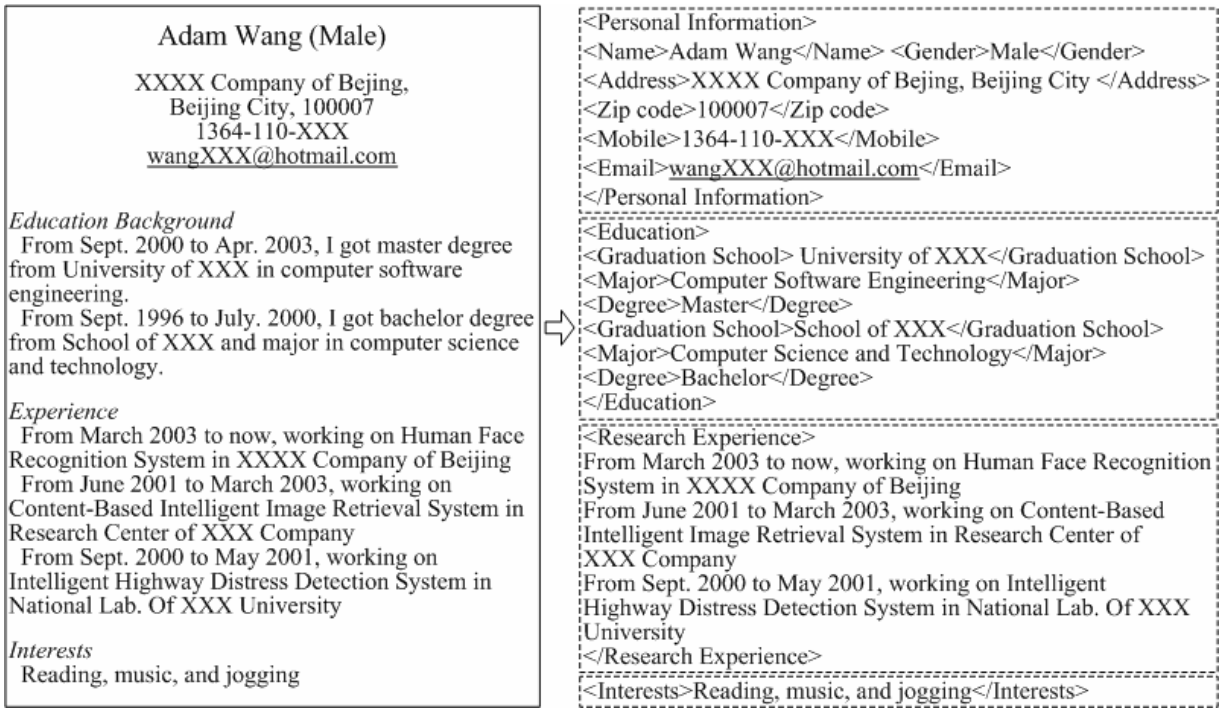

Figure 1. Example of a resume and the extracted information.

constituting a restricted domain, resumes can be written in multitude of formats (e.g. structured tables or plain texts), in different languages (e.g. Chinese and English) and in different file types (e.g. Text, PDF, Word etc.). Moreover, writing styles could be very diversified.

Among the methods in IE, Hidden Markov modelling has been widely used (Freitag and McCallum, 1999; Borkar et al., 2001). As a state-based model, HMMs are good at extracting information fields that hold a strong order of sequence. Classification is another popular method in IE. By assuming the independence of information types, it is feasible to classify segmented units as either information types to be extracted (Kushmerick et al., 2001; Peshkin and Pfeffer, 2003; Sitter and Daelemans, 2003), or information boundaries (Finn and Kushmerick, 2004). This method specializes in settling the extraction problem of independent information types.

Resume shares a document-level hierarchical contextual structure where the related information units usually occur in the same textual block, and text blocks of different information categories usually occur in a relatively fixed order. Such characteristics have been successfully used in the categorization of multi-page documents by Frasconi et al. (2001).

In this paper, given the hierarchy of resume information, a cascaded two-pass IE framework is designed. In the first pass, the general information is extracted by segmenting the entire resume into consecutive blocks and each block is annotated with a label indicating its category. In the second pass, detailed information pieces are further extracted within the boundary of certain blocks. Moreover, for different types of information, the most appropriate extraction method is selected through experiments. For the first pass, since there exists a strong sequence among blocks, a HMM model is applied to segment a resume and each block is labelled with a category of general information. We also apply HMM for the educational detailed information extraction for the same reason. In addition, classification based method is selected for the personal detailed information extraction where information items appear relatively independently.

Tested with 1,200 Chinese resumes, experimental results show that exploring the hierarchical structure of resumes with this proposed cascaded framework improves the average F-score of detailed information extraction

greatly, and combining different IE models in different layer properly is effective to achieve good precision and recall.

The remaining part of this paper is structured as follows. Section 2 introduces the related work. Section 3 presents the structure of the cascaded hybrid IE model and introduces the HMM model and SVM model in detail. Experimental results and analysis are shown in Section 4. Section 5 provides a discussion of our cascaded hybrid model. Section 6 is the conclusion and future work.

2 Related Work

As far as we know, there are few published works on resume IE except some products, for which there is no way to determine the technical details. One of the published results on resume IE was shown in Ciravegna and Lavelli (2004). In this work, they applied $(LP)^2$, a toolkit of IE, to learn information extraction rules for resumes written in English. The information defined in their task includes a flat structure of *Name*, *Street*, *City*, *Province*, *Email*, *Telephone*, *Fax* and *Zip code*. This flat setting is not only different from our hierarchical structure but also different from our detailed information pieces.

Besides, there are some applications that are analogous to resume IE, such as seminar announcement IE (Freitag and McCallum, 1999), job posting IE (Sitter and Daelemans, 2003; Finn and Kushmerick, 2004) and address segmentation (Borkar et al., 2001; Kushmerick et al., 2001). Most of the approaches employed in these applications view a text as flat and extract information from all the texts directly (Freitag and McCallum, 1999; Kushmerick et al., 2001; Peshkin and Pfeffer, 2003; Finn and Kushmerick, 2004). Only a few approaches extract information hierarchically like our model. Sitter and Daelemans (2003) present a double classification approach to perform IE by extracting words from pre-extracted sentences. Borkar et al. (2001) develop a nested model, where the outer HMM captures the sequencing relationship among elements and the inner HMMs learn the finer structure within each element. But these approaches employ the same IE methods for all the information types. Compared with them, our model applies different methods in different sub-tasks to fit the special contextual structure of information in each sub-task well.

3 Cascaded Hybrid Model

Figure 2 is the structure of our cascaded hybrid model. The first pass (on the left hand side) segments a resume into consecutive blocks with a HMM model. Then based on the result, the second pass (on the right hand side) uses HMM to extract the educational detailed information and SVM to extract the personal detailed information, respectively. The block selection module is used to decide the range of detailed information extraction in the second pass.

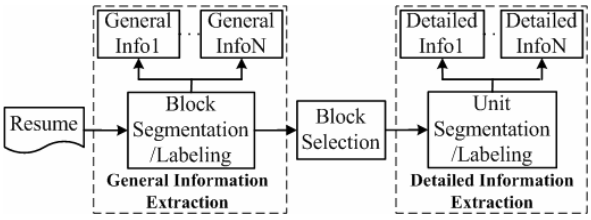

Figure 2. Structure of cascaded hybrid model.

3.1 HMM Model

3.1.1 Model Design

For general information, the IE task is viewed as labelling the segmented units with predefined class labels. Given an input resume T which is a sequence of words $w_1,w_2,\ldots,w_k$, the result of general information extraction is a sequence of blocks in which some words are grouped into a certain block $T = t_1, t_2,\ldots, t_n$, where t_i is a block. Assuming the expected label sequence of T is $L=l_1, l_2,\ldots, l_n$, with each block being assigned a label l_i, we get the sequence of block and label pairs $Q=(t_1, l_1), (t_2, l_2),\ldots,(t_n, l_n)$. In our research, we simply assume that the segmentation is based on the natural paragraph of T.

Table 1 gives the list of information types to be extracted, where general information is represented as G_1~G_7. For each kind of general information, say G_i, two labels are set: G_i-*B* means the beginning of G_i, G_i-*M* means the remainder part of G_i. In addition, label *O* is defined to represent a block that does not belong to any general information types. With these positional information labels, general information can be obtained. For instance, if the label sequence Q for

a resume with 10 paragraphs is $Q=(t_1, G_1\text{-}B)$, $(t_2, G_1\text{-}M)$, $(t_3, G_2\text{-}B)$, $(t_4, G_2\text{-}M)$, $(t_5, G_2\text{-}M)$, (t_6, O) , (t_7, O) , $(t_8, G_3\text{-}B)$, $(t_9, G_3\text{-}M)$, $(t_{10}, G_3\text{-}M)$, three types of general information can be extracted as follows: G_1:$[t_1, t_2]$, G_2:$[t_3, t_4, t_5]$, G_3:$[t_8, t_9, t_{10}]$.

Formally, given a resume $T=t_1,t_2,\ldots,t_n$, seek a label sequence $L^*=l_1,l_2,\ldots,l_n$, such that the probability of the sequence of labels is maximal.

$$L^* = \arg\max_L P(L \mid T) \quad (1)$$

According to Bayes' equation, we have

$$L^* = \arg\max_L P(T \mid L) \times P(L) \quad (2)$$

If we assume the independent occurrence of blocks labelled as the same information types, we have

$$P(T \mid L) = \prod_{i=1}^{n} P(t_i \mid l_i) \quad (3)$$

We assume the independence of words occurring in t_i and use a unigram model, which multiplies the probabilities of these words to get the probability of t_i.

$$P(t_i \mid l_i) = \prod_{r=1}^{m} P(w_r \mid l_i), \text{where } t_i = \{w_1, w_2, \ldots w_m\} \quad (4)$$

If a tri-gram model is used to estimate $P(L)$, we have

$$P(L) = P(l_1) \times P(l_2 \mid l_1) \prod_{i=3}^{n} P(l_i \mid l_{i-1}, l_{i-2}) \quad (5)$$

To extract educational detailed information from *Education* general information, we use another HMM. It also uses two labels D_i-B and D_i-M to represent the beginning and remaining part of D_i, respectively. In addition, we use label O to represent that the corresponding word does not belong to any kind of educational detailed information. But this model expresses a text T as word sequence $T=w_1,w_2,\ldots,w_n$. Thus in this model, the probability $P(L)$ is calculated with Formula 5 and the probability $P(T|L)$ is calculated by

$$P(T \mid L) = \prod_{i=1}^{n} P(w_i \mid l_i) \quad (6)$$

Here we assume the independent occurrence of words labelled as the same information types.

3.1.2 Parameter Estimation

Both words and named entities are used as features in our HMMs. A Chinese resume $C= c_1', c_2', \ldots, c_k'$ is first tokenized into $C= w_1, w_2, \ldots, w_k$ with a Chinese word segmentation system LSP (Gao et al., 2003). This system outputs predefined features, including words and named entities in 8 types (Name, Date, Location, Organization, Phone, Number, Period, and Email). The named entities of the same type are normalized into single ID in feature set.

In both HMMs, fully connected structure with one state representing one information label is applied due to its convenience. To estimate the probabilities introduced in 3.1.1, maximum likelihood estimation is used, which are

$$P(l_i \mid l_{i-1}, l_{i-2}) = \frac{count(l_i, l_{i-1}, l_{i-2})}{count(l_{i-1}, l_{i-2})} \quad (7)$$

$$P(l_i \mid l_{i-1}) = \frac{count(l_i, l_{i-1})}{count(l_{i-1})} \quad (8)$$

$$P(w_r \mid l_i) = \frac{count(w_r, l_i)}{\sum_{r=1}^{m} count(w_r, l_i)}, \quad (9)$$

where state i contains m distinct words

3.1.3 Smoothing

Short of training data to estimate probability is a big problem for HMMs. Such problems may occur when estimating either $P(T|L)$ with unknown word w_i or $P(L)$ with unknown events.

Bikel et al. (1999) mapped all unknown words to one token _UNK_ and then used a held-out data to train the bi-gram models where unknown words occur. They also applied a back-off strategy to solve the data sparseness problem when estimating the context model with unknown events, which interpolates the estimation from training corpus and the estimation from the back-off model with calculated parameter λ (Bikel et al., 1999). Freitag and McCallum (1999) used shrinkage to estimate the emission probability of unknown words, which combines the estimates from data-sparse states of the complex model and the estimates in related data-rich states of the simpler models with a weighted average.

In our HMMs, we first apply Good Turing smoothing (Gale, 1995) to estimate the probability $P(w_r|l_i)$ when training data is sparse. For word w_r seen in training data, the emission probability is $P(w_r|l_i)\times(1\text{-}x)$, where $P(w_r|l_i)$ is the emission probability calculated with Formula 9 and $x=E_i/S_i$ (E_i is the number of words appearing only once in state i and S_i is the total number of words occurring in state i). For unknown word w_r, the emission probability is $x/(M\text{-}m_i)$, where M is the number of all the words appearing in training data,

and m_i is the number of distinct words occurring in state i. Then, we use a back-off schema (Katz, 1987) to deal with the data sparseness problem when estimating the probability $P(L)$ (Gao et al., 2003).

3.2 SVM Model

3.2.1 Model Design

We convert personal detailed information extraction into a classification problem. Here we select SVM as the classification model because of its robustness to over-fitting and high performance (Sebastiani, 2002). In the SVM model, the IE task is also defined as labelling segmented units with predefined class labels. We still use two labels to represent personal detailed information P_i: P_i-B represents the beginning of P_i and P_i-M represents the remainder part of P_i. Besides of that, label O means that the corresponding unit does not belong to any personal detailed information boundaries and information types. For example, for part of a resume "Name:Alice (Female)", we got three units after segmentation with punctuations, i.e. "Name", "Alice", "Female". After applying SVM classification, we can get the label sequence as P_1-B,P_1-M,P_2-B. With this sequence of unit and label pairs, two types of personal detailed information can be extracted as P_1: [Name:Alice] and P_2: [Female].

Various ways can be applied to segment T. In our work, segmentation is based on the natural sentence of T. This is based on the empirical observation that detailed information is usually separated by punctuations (e.g. comma, Tab tag or Enter tag).

The extraction of personal detailed information can be formally expressed as follows: given a text $T=t_1,t_2,\ldots,t_n$, where t_i is a unit defined by the segmenting method mentioned above, seek a label sequence $L^* = l_1,l_2,\ldots,l_n$, such that the probability of the sequence of labels is maximal.

$$L^* = \arg\max_{L} P(L \mid T) \qquad (10)$$

The key assumption to apply classification in IE is the independence of label assignment between units. With this assumption, Formula 10 can be described as

$$L^* = \arg\max_{L=l_1,l_2\ldots l_n} \prod_{i=1}^{n} P(l_i \mid t_i) \qquad (11)$$

Thus this probability can be maximized by maximizing each term in turn. Here, we use the SVM score of labelling t_i with l_i to replace $P(l_i|t_i)$.

3.2.2 Multi-class Classification

SVM is a binary classification model. But in our IE task, it needs to classify units into N classes, where N is two times of the number of personal detailed information types. There are two popular strategies to extend a binary classification task to N classes (A.Berger, 1999). The first is *One vs. All* strategy, where N classifiers are built to separate one class from others. The other is *Pairwise* strategy, where $N\times(N\text{-}1)/2$ classifiers considering all pairs of classes are built and final decision is given by their weighted voting. In our model, we apply the *One vs. All* strategy for its good efficiency in classification. We construct one classifier for each type, and classify each unit with all these classifiers. Then we select the type that has the highest score in classification. If the selected score is higher than a predefined threshold, then the unit is labelled as this type. Otherwise it is labelled as O.

3.2.3 Feature Definition

Features defined in our SVM model are described as follows:

Word: Words that occur in the unit. Each word appearing in the dictionary is a feature. We use $TF\times IDF$ as feature weight, where TF means word frequency in the text, and IDF is defined as:

$$IDF(w) = Log_2 \frac{N}{N_w} \qquad (12)$$

N: the total number of training examples;
N_w: the total number of positive examples that contain word w

Named Entity: Similar to the HMM models, 8 types of named entities identified by LSP, i.e., Name, Date, Location, Organization, Phone, Number, Period, Email, are selected as binary features. If any one type of them appears in the text, then the weight of this feature is 1, otherwise is 0.

3.3 Block Selection

Block selection is used to select the blocks generated from the first pass as the input of the second pass for detailed information extraction.

Error analysis of preliminary experiments shows that the majority of the mistakes of general information extraction resulted from labelling non-

Model	Personal Detailed Info (SVM)			Educational Detailed Info (HMM)		
	Avg.P (%)	Avg.R (%)	Avg.F (%)	Avg.P (%)	Avg.R (%)	Avg.F (%)
Flat	77.49	82.02	77.74	58.83	77.35	66.02
Cascaded	**86.83** (+9.34)	**76.89** (-5.13)	**80.44** (+2.70)	**70.78** (+11.95)	**76.80** (-0.55)	**73.40** (+7.38)

Table 2. IE results with cascaded model and flat model.

boundary blocks as boundaries in the first pass. Therefore we apply a fuzzy block selection strategy, which not only selects the blocks labelled with target general information, but also selects their neighboring two blocks, so as to enlarge the extracting range.

4 Experiments and Analysis

4.1 Data and Experimental Setting

We evaluated this cascaded hybrid model with 1,200 Chinese resumes. The data set was divided into 3 parts: training data, parameter tuning data and testing data with the proportion of 4:1:1. 6-folder cross validation was conducted in all the experiments. We selected SVMlight (Joachims, 1999) as the SVM classifier toolkit and LSP (Gao et al., 2003) for Chinese word segmentation and named entity identification. Precision (P), recall (R) and F-score (F=2PR/(P+R)) were used as the basic evaluation metrics and macro-averaging strategy was used to calculate the average results. For the special application background of our resume IE model, the "Overlap" criterion (Lavelli et al., 2004) was used to match reference instances and extracted instances. We define that if the proportion of the overlapping part of extracted instance and reference instance is over 90%, then they match each other.

A set of experiments have been designed to verify the effectiveness of exploring document-level hierarchical structure of resume and choose the best IE models (HMM vs. classification) for each sub-task.

- Cascaded model vs. flat model

Two flat models with different IE methods (SVM and HMM) are designed to extract personal detailed information and educational detailed information respectively. In these models, no hierarchical structure is used and the detailed information is extracted from the entire resume texts rather than from specific blocks. These two flat models will be compared with our proposed cascaded model.

- Model selection for different IE tasks

Both SVM and HMM are tested for all the IE tasks in first pass and in second pass.

4.2 Cascaded Model vs. Flat Model

We tested the flat model and cascaded model with detailed information extraction to verify the effectiveness of exploring document-level hierarchical structure. Results (see Table 2) show that with the cascaded model, the precision is greatly improved compared with the flat model with identical IE method, especially for educational detailed information. Although there is some loss in recall, the average F-score is still largely improved in the cascaded model.

4.3 Model Selection for Different IE Tasks

Then we tested different models for the general information and detailed information to choose the most appropriate IE model for each sub-task.

Model	Avg.P (%)	Avg.R (%)
SVM	80.95	72.87
HMM	75.95	75.89

Table 3. General information extraction with different models.

Model	Personal Detailed Info		Educational Detailed Info	
	Avg.P (%)	Avg.R (%)	Avg.P (%)	Avg.R (%)
SVM	86.83	76.89	67.36	66.21
HMM	79.64	60.16	70.78	76.80

Table 4. Detailed information extraction with different models.

Results (see Table 3) show that compared with SVM, HMM achieves better recall. In our cascaded framework, the extraction range of detailed information is influenced by the result of general information extraction. Thus better recall of general information leads to better recall of detailed information subsequently. For this reason,

we choose HMM in the first pass of our cascaded hybrid model.

Then in the second pass, different IE models are tested in order to select the most appropriate one for different sub-tasks. Results (see Table 4) show that HMM performs much better in both precision and recall than SVM for educational detailed information extraction. We think that this is reasonable because HMM takes into account the sequence constraints among educational detailed information types. Therefore HMM model is selected to extract educational detailed information in our cascaded hybrid model. While for the personal detailed information extraction, we find that the SVM model gets better precision and recall than HMM model. We think that this is because of the independent occurrence of personal detailed information. Therefore, we select SVM to extract personal detailed information in our cascaded model.

5 Discussion

Our cascaded framework is a "pipeline" approach and it may suffer from error propagation. For instance, the error in the first pass may be transferred to the second pass when determining the extraction range of detailed information. Therefore the precision and recall of detailed information extraction in the second pass may be decreased subsequently. But we are not sure whether N-Best approach (Zhai et al., 2004) would be helpful. Because our cascaded hybrid model applies different IE methods for different sub-tasks, it is difficult to incorporate the N-best strategy by either simply combining the scores of the first pass and the second pass, or using the scores of the second pass to do re-ranking to select the best results. Instead of using N-best, we apply a fuzzy block selection strategy to enlarge the search scope. Experimental results of personal detailed information extraction show that compared with the exact block selection strategy, this fuzzy strategy improves the average recall of personal detailed information from 68.48% to 71.34% and reduce the average precision from 83.27% to 81.71%. Therefore the average F-score is improved by the fuzzy strategy from 75.15% to 76.17%.

Features are crucial to our SVM model. For some fields (such as *Name*, *Address* and *Graduation School*), only using words as features may result in low accuracy in IE. The named entity (NE) features used in our model enhance the accuracy of detailed information extraction. As exemplified by the results (see Table 5) on personal detailed information extraction, after adding named entity features, the F-score are improved greatly.

Field	Word +NE (%)	Word (%)
Name	90.22	3.11
Birthday	87.31	84.82
Address	67.76	49.16
Phone	81.57	75.31
Mobile	70.64	58.01
Email	88.76	85.96
Registered Residence	75.97	72.73
Residence	51.61	42.86
Graduation School	40.96	15.38
Degree	73.20	63.16
Major	63.09	43.24

Table 5. Personal detailed information extraction with different features (Avg.F).

In our cascaded hybrid model, we apply HMM and SVM in different pass separately to explore the contextual structure of information types. It guarantees the simplicity of our hybrid model. However, there are other ways to combine state-based and discriminative ideas. For example, Peng and McCallum (2004) applied Conditional Random Fields to extract information, which draws together the advantages of both HMM and SVM. This approach could be considered in our future experiments.

Some personal detailed information types do not achieve good average F-score in our model, such as *Zip code* (74.50%) and *Mobile* (73.90%). Error analysis shows that it is because these fields do not contain distinguishing words and named entities. For example, it is difficult to extract *Mobile* from the text "Phone: 010-62617711 (13859750123)". But these fields can be easily distinguished with their internal characteristics. For example, *Mobile* often consists of certain length of digital figures. To identify these fields, the Finite-State Automaton (FSA) that employs hand-crafted grammars is very effective (Hsu and Chang, 1999). Alternatively, rules learned from annotated data are also very promising in handling this case (Ciravegna and Lavelli, 2004).

We assume the independence of words occurring in unit t_i to calculate the probability

$P(t_i|l_i)$ in HMM model. While in Bikel et al. (1999), a bi-gram model is applied where each word is conditioned on its immediate predecessor when generating words inside the current name-class. We will compare this method with our current method in the future.

6 Conclusions and Future Work

We have shown that a cascaded hybrid model yields good results for the task of information extraction from resumes. We tested different models for the first pass and the second pass, and for different IE tasks. Our experimental results show that the HMM model is effective in handling the general information extraction and educational detailed information extraction, where there exists strong sequence of information pieces. And the SVM model is effective for the personal detailed information extraction.

We hope to continue this work in the future by investigating the use of other well researched IE methods. As our future works, we will apply FSA or learned rules to improve the precision and recall of some personal detailed information (such as *Zip code* and *Mobile*). Other smoothing methods such as (Bikel et al. 1999) will be tested in order to better overcome the data sparseness problem.

7 Acknowledgements

The authors wish to thank Dr. JianFeng Gao, Dr. Mu Li, Dr. Yajuan Lv for their help with the LSP tool, and Dr. Hang Li, Yunbo Cao for their valuable discussions on classification approaches. We are indebted to Dr. John Chen for his assistance to polish the English. We want also thank Long Jiang for his assistance to annotate the training and testing data. We also thank the three anonymous reviewers for their valuable comments.

References

A.Berger. Error-correcting output coding for text classification. 1999. In *Proceedings of the IJCAI-99 Workshop on Machine Learning for Information Filtering*.

D.M.Bikel, R.Schwartz, R.M.Weischedel. 1999. An algorithm that learns what's in a name. *Machine Learning*, 34(1):211-231.

V.Borkar, K.Deshmukh and S.Sarawagi. 2001. Automatic segmentation of text into structured records. In *Proceedings of ACM SIGMOD Conference*. pp.175-186.

F.Ciravegna, A.Lavelli. 2004. LearningPinocchio: adaptive information extraction for real world applications. *Journal of Natural Language Engineering*, 10(2):145-165.

A.Finn and N.Kushmerick. 2004. Multi-level boundary classification for information extraction. In *Proceedings of ECML04*.

P.Frasconi, G.Soda and A.Vullo. 2001. Text categorization for multi-page documents: a hybrid Naïve Bayes HMM approach. In *Proceedings of the 1st ACM/IEEE-CS Joint Conference on Digital Libraries*. pp.11-20.

D.Freitag and A.McCallum. 1999. Information extraction with HMMs and shrinkage. In *AAAI99 Workshop on Machine Learning for Information Extraction*. pp.31-36.

W.Gale. 1995. Good-Turing smoothing without tears. *Journal of Quantitative Linguistics*, 2:217-237.

J.F.Gao, M.Li and C.N.Huang. 2003. Improved source-channel models for Chinese word segmentation. In *Proceedings of ACL03*. pp.272-279.

C.N.Hsu and C.C.Chang. 1999. Finite-state transducers for semi-structured text mining. In *Proceedings of IJCAI99 Workshop on Text Mining: Foundations, Techniques and Applications*. pp.38-49.

T.Joachims. 1999. Making large-scale SVM learning practical. *Advances in Kernel Methods - Support Vector Learning*. MIT-Press.

S.M.Katz. 1987. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE ASSP*, 35(3):400-401.

N.Kushmerick, E.Johnston and S.McGuinness. 2001. Information extraction by text classification. In *IJCAI01 Workshop on Adaptive Text Extraction and Mining*.

A.Lavelli, M.E.Califf, F.Ciravegna, D.Freitag, C.Giuliano, N.Kushmerick and L.Romano. 2004. A critical survey of the methodology for IE evaluation. In *Proceedings of the 4th International Conference on Language Resources and Evaluation*.

F.Peng and A.McCallum. 2004. Accurate information extraction from research papers using conditional random fields. In *Proceedings of HLT/NAACL-2004*. pp.329-336.

L.Peshkin and A.Pfeffer. 2003. Bayesian information extraction network. In *Proceedings of IJCAI03*. pp.421-426.

F.Sebastiani. 2002. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1-47.

A.D.Sitter and W.Daelemans. 2003. Information extraction via double classification. In *Proceedings of ATEM03*.

L.Zhai, P.Fung, R.Schwartz, M.Carpuat and D.Wu. 2004. Using N-best lists for named entity recognition from Chinese speech. In *Proceedings of HLT/NAACL-2004*.