2017 9th International Conference on Knowledge and Systems Engineering(KSE)

Intent Extraction from Social Media Texts Using Sequential Segmentation and Deep Learning Models

Thai-Le Luong*, Minh-Son Cao†, Duc-Thang Le† and Xuan-Hieu Phan†
*University of Transport and Communications, Hanoi, Vietnam
†University of Engineering and Technology, Vietnam National University, Hanoi
Email: luongthaile80@utc.edu.vn, {soncm_58, thangld_58, hieupx}@vnu.edu.vn

***Abstract*—Nowadays, users are much more willing to share their daily activities, their thoughts or feelings, and even their intentions (e.g., *buy an apartment*, *rent a car*, *travel to somewhere*, etc.) on online social media channels. Understanding intents of online users, therefore, has become a crucial need in many different business areas like production, finance/banking, real estate, tourism, e–commerce, and online marketing. In this paper, we will present our solutions to extract intent information from online social media texts. This can be seen as an information extraction or sequential segmentation task. In order to perform this task, we have built our machine learning models based on conditional random fields (CRFs), an advanced statistical graphical model for sequence data, and bidirectional long short–term memory (Bi–LSTM), a well–known deep learning model. To evaluate our methods, we have defined two intent extraction tasks in two domains: real estate and cosmetic & beauty. For each task, we have defined the tag set as well as prepared the labeled data set that consists of Vietnamese social media text posts with the annotation of intent words/phrases. Experimental results showed that the proposed methods can effectively extract intent information from online texts with significantly high accuracy.**

***Index Terms*—Intention mining, intent extraction, user intent identification, information extraction, text mining.**

I. Introduction

Fully understanding of user intents from texts, in general or open domain, is actually a natural language understanding problem, and it is extremely hard to tackle. However, it can be solved gradually by narrowing its scope and domain and dividing into multiple steps. In our previous work [16], we have defined a three–stage process for user intent identification that includes three main tasks: (1) user intent filtering, (2) intent domain identification, and (3) intent parsing and extraction. The first two tasks have been addressed in our previous work as normal classification problems [16], [17]. This study focuses on the third task. That is, given a text post carrying an intent and its domain, building machine learning models to parse and extract intent information from the post. The intent information consists of two parts: **intent head** (including *intent name* and *object*) and **intent properties** or **constraints** (including all possible properties, constraints, or parameters of an intent like *brand*, *location*, *price*, etc.). With different text domains, intent information can have different sets of labels. And the difference is normally about intent properties. For instance, an intent in **tourism** domain should have properties like *destination*, *time/duration*, *number of people* . . . while an intent about **real estate** should have *apartment type*, *number of rooms*, *area*, etc. Technically, parsing and extracting intent information is an information extraction (IE) problem where the label set can vary depending on the text domain.

In this paper, we formulate intent extraction as a sequential segmentation problem. For a particular text domain, we will define a tag/label set, prepare an annotated data set, and and then build extraction models using existing machine learning techniques to extract intent head and properties. For example, with a post in **cosmetic** & **beauty** domain like "*I want to have my eye muscle orthopedic operated naturally according to the WONJIN form*", the resulting intent information should be *intent name*="*orthopedic operated*" and its properties are "*eye muscle*", "*naturally*", and "*WONJIN form*". As stated earlier, it is hard to perform open–domain intent extraction because the label sets vary a lot with different domains. We, on the other hand, perform intent extraction for two particular domains: **cosmetic** & **beauty** and **real estate**. These two domains have their own label sets shown in Table I and II. We have also annotated two medium–sized data sets for the two domains for evaluation.

In order to perform intent extraction from text posts, we chose to use two advanced machine learning techniques that should be suitable for segmenting sequence data. The first is conditional random fields (CRFs) [14], a powerful statistical graphical model that was designed to labeling and segmenting sequence data. The second is long short–term memory (LSTM) [8], a deep learning model that has been widely and successfully used for sentence–level natural language processing tasks like named entity recognition, language modeling since they can integrate long temporal dependency information.

As stated earlier, user intent understanding is a very difficult NLP task to solve. However, in this work, we have simplified and narrowed its scope/domain in order to solve it using existing supervised machine learning techniques. Technically, in this paper, we have several major contributions as follows. (1) To the best of our knowledge, our work is the first attempt to address intent extraction for Vietnamese texts. (2) We proposed the definition of tag or label sets for intent information in which *intent name* and *intent object* are domain–independent and intent properties or constraints are different for each domain. (3) We have built two medium–sized corpora for the two domains: **cosmetic** & **beauty** and **real estate**. And (4) we have proposed and investigated the use of advanced statistical machine learning methods (CRFs and LSTM) for solving this

978-1-5386-3576-6/17/$31.00

task. The experimental results of the two domains showed that this task can be solved with significantly high accuracy.

II. PREVIOUS WORK

Almost all previous studies considered the problem of identifying user intent as a classification task [2], [5], [9], where they tried to classify user queries or text posts in into a particular class such as *purchase intent* or *non–purchase intent* [5]. Meanwhile, some of them approached the problem using clustering methods [12], in which the they clustered queries with similar intent and furthermore, they leveraged the discovered intent patterns to automatically annotate a larger number of queries. Besides, there were several studies proposing graph based methods to learn the user intents [19], [21], [23], [24]. For example, Wang et al. (2015) [24] took advantage of effective information propagation via graph regularization to label intention tweets. Semantic parsing is also an approach to predict the users intents [11], [13], [18], [25] and it seems to be similar to ours. Li et al. (2010) [18] formally defined the semantic structure of noun phrase queries is comprised of intent heads (IH) and intent modifiers (IM), where an IH of a query represents the essential information the user intends to obtain and the IMs serve as the filters of the information the user receives. Especially in recent years, we found several new studies in intent detection using deep neural network technique [6], [13]. However, they only focus on queries classification or inferring intention from the spoken language using semantically enriched word embedding. It is quite different from ours where we attempt to deeply understand the user intention by detecting the intent head (keywords) and other intent properties or constraints. Our work may be considered as an information extraction task. Several other neural architectures have previously been proposed for NER, that seems to be similar to ours. For example, Collobert et al. (2011) [3] uses a CNN over a sequence of word embeddings with a CRF layer on top. More recently, Huang et al. (2015) [10] presented a model similar to our LSTM-CRF, but using hand-crafted spelling features. And the most relevant to our approach architecture is the research of Lample et all (2016) [15]. They claimed that, their experiments obtained state-of-the-art NER performance in Dutch, German, Spainish.

III. INTENT IDENTIFICATION AND INFORMATION EXTRACTION

According to our previous work [16], a user intent is defined as a quintuple (5–tuple) as follows:

$$\mathbf{I}_u^e = \langle u, \mathbf{c}, d, w, \mathbf{p} \rangle \tag{1}$$

in which:

- u is the user identifier on social media services.
- $\mathbf{c}$ is the current context or condition around this intent. For example, a user may currently be pregnant, sick, or having baby. Context $\mathbf{c}$ also includes the time at which the intent was expressed or posted on online.
- d is the intent domain such as *real-estate*, *finance and banking*, *education*, etc.

Domain	Sentences extracted from posts	Intent key-word (w)	Information extracted (p)
Cosmetic & Beauty	Em muốn chỉnh hình cơ mắt tự nhiên của WONJIN	**chỉnh hình** (orthopedic operate)	*Object*: cơ mắt (eye muscle) *Type*: tự nhiên (naturally) *Brand*: WONJIN
	I want to have my eye's muscle orthopedic operate naturally according to the WONJIN form		
Real-Estate	Bán nhà ngõ rộng phố Dương Quảng Hàm, phường Quan Hoa, quận Cầu Giấy, ngay đầu ngõ, 5 tầng, diện tích 4x17,1m, vị trí đẹp, tiện buôn bán, mở văn phòng, giá thỏa thuận, miễn trung gian. Tel. 0968562689	**bán** (sell)	*Object*: nhà (house) *Acreage*: 4x17,1m *Location*: phố (street) Dương Quảng Hàm, phường (ward) Quan Hoa, quận (district) Cầu Giấy *Number of floors*: 5 *Seller contact*: 0968562689
	For sale: a house with large lane at Duong Quang Ham street, Quan Hoa ward, Cau Giay district, 5 floor, the acreage is 4x17,1m…Tel 0968562689		

Figure 1. Examples of intent name (intent keyword or phrase), intent objects and their properties extracted from online texts

- w is the intent name, i.e., a keyword or phrase representing the intent. It may be the name of a thing or an action of interest. For example, w can be *rent* (house), *borrow* (loan), or *orthopedic–operate* (eye), etc.
- $\mathbf{p}$ is a list of properties or constraints associated with an intent. It is a list of property–value pairs related to the intent. For example, $\mathbf{p}$ can be {location="*Duong Quang Ham street* …", number–of–floors="*5*", …}.

In this paper, we focus on detecting the two elements w, and $\mathbf{p}$ in the formula (1), that is, to identify the intent name and intent properties of an intention text post. We also choose to perform intent extraction with two domains: **real estate** and **cosmetic** & **beauty**. The Figure 1 shows some examples of the online texts with corresponding w and $\mathbf{p}$ be extracted.

IV. MAIN ARCHITECTURES

A. LSTM Architecture

LSTM was introduced by Hochreiter and Schmidhuber (1997) [8], and it is based on recurrent neural network (RNN) architecture with the desire to capture the long–term dependencies. Given the input $(x_1, x_2, ..., x_n)$, we have LSTM model computes the state sequence $(h_1, h_2, ..., h_n)$ by iteratively applying the following updates:

$$i_t = \sigma(W_x^{(i)} x_t + W_h^{(i)} h_{t-1} + W_c^{(i)} c_{t-1} + b^{(i)})$$
$$c_t = (1 - i_t) \odot c_{t-1} + i_t \odot tanh(W_x^{(c)} x_t + W_h^{(c)} h_{t-1} + b^{(c)})$$
$$o_t = \sigma(W_x^{(o)} x_t + W_h^{(o)} h_{t-1} + W_c^{(o)} c_t + b^{(o)})$$
$$h_t = o_t \odot tanh(c_t)$$

Where σ is the element–wise sigmoid function and $\odot$ is the element–wise product. Long short–term memory have been designed to combat RNN issue by incorporating a memory–cell and have been shown to capture long–range dependencies. They do so using several gates that control the proportion of the input to give to the memory cell, and the proportion from the previous state to forget.

For the given sentence $(x_1, x_2, ..., x_n)$ containing n words, each represented as a d–dimensional vector, an LSTM, which

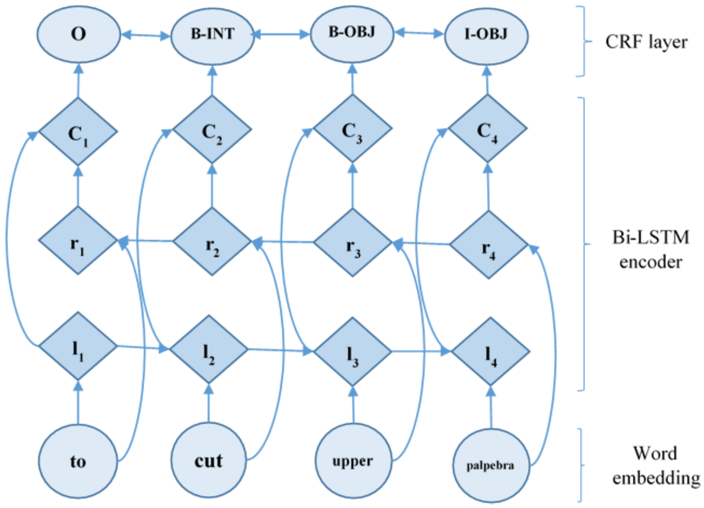

Figure 2. Main architecture of the network

refer to the forward LSTM, computes a representation h_t^l of the left context of the sentence at every word t. Naturally, a backward LSTM model generates a representation of the right context h_t^r as well as should add useful information. This forward and backward LSTM pair is referred to as a bidirectional LSTM [4], that will be applied to our model. This Bi–LSTM model will be shown concretely in Figure 2.

B. CRFs for Segmenting and Labeling Sequense Data

Conditional random fields CRFs (Lafferty et al., 2001) [14] has shown a great success in sequence labeling problem. Therefore, we propose the use of CRFs to our intent extraction problem. CRF model is a conditional probability of output label sequence given the input data observation. In our task, the input data sequence is a text post where each word is an observation, and the output label sequence is a list of labels that will indicate which words are intent name, intent object, and other properties. Training CRF models require annotated training data, i.e., with intent information marked. As stated earlier, in this paper, we will build CRFs extraction models with two domains: **real estate** and **cosmetic** & **beauty**.

C. LSTM-CRF Model

Traditional LSTM tagging model use the h_t as features to make independent tagging decision for each output y_t [20]. However, in this work, we used a different tagging model that was published in 2016 [15]. It is called LSTM–CRF tagging model. Its main architecture is that word embedding are given to a Bi–LSTM. l_i represents the word i and its left context, r_i represents the word i and its right context. These two vectors will be combined to obtain the result of the word i in its context, c_i. The idea of this model is very simple but surprisingly effective. Instead of making tagging independently, they add a CRF layer at the end of the tagging processing, where the output of LSTM layer had been considered as the input of CRF layer and the output of CRF layer will be the final tags. The Figure 2 shows the main architecture of the model.

V. Experiments

A. Building The Label Sets

After looking at the crawled data, we built the two sets of labels for the two domains. Because in this work, we only focus on identifying intent name, intent object and intent properties, so we pay attention to several types of labels for each domain. It turns out not easy because in online texts, there are too much information relating to user intent. But, some of the information are not really important and even hardly seen in online texts. We, therefore, have to decide what information to extract or not. Finally, we have 13 labels for **real estate** domain and 9 labels for **cosmetic** & **beauty** domain. Those labels are described in Table II and Table I below.

B. Experimental Data

To build our own experimental data, we automatically crawled the data from several popular websites and forums in Vietnam, such as *muaban.net*, *diendanthammy.net*, *vatgia.com*, *webtretho.com*, etc. In order to reduce noise, we only kept text posts containing more than 30 characters and less than 800 characters. We collected totally 712 posts for **real estate** and 1500 posts for **cosmetic** & **beauty**. We then asked a group of three students to label this data according to a clear labeling guideline with the label sets described in Table II and Table I. The annotation is chosen based on the majority of tagging results among students. As mentioned above, we built two different models to conduct the experiments, which are LSTM–CRF model and CRF itself. For LSTM–CRF, the labeled data in each domain were randomly divided into three parts with the proportion 3:1:1. In which, the first part was used for training, the second for validation, and the last for testing. We trained the model by using training set, tuning the parameters using validation set, and then we tested the performance on the test set. For CRFs model, we used the same training set and test set, and the validation set was excluded.

C. LSTM Technique Experiment with Three Options

1) Character-based Model: This is to build LSTM model using character–level features instead of just learning based on words. Learning character–embedding has many advantages. They are useful for morphologically rich languages [20]. First, the model initializes randomly a character lookup table containing embedding information for all characters. Then, the embedding vector corresponding to every character in a word will be given in direct and reverse order to a forward–backward LSTM. After concatenating the forward and the backward representations from the bidirectional LSTM, we derive the embedding for a word. Finally, this character–level representation is paired with a word–level representation from a word lookup table. As a result, we expect the final forward LSTM representation to be a more accurate representation of the suffix of the word, and the final backward LSTM result to be a better representation of its prefix.

TABLE I
THE LIST OF 9 LABELS FOR THE COSMETIC & BEAUTY DOMAIN

Label name	Descriptions	Label	#
Intent	Intent keyword of the online text/user	int	1944
Object	The object that the user's intent refers to	obj	1799
Age	The age of the client	age	131
Brand	The brand of the service/product/object that be referred to in the online text	bra	70
Duration of illness	The duration time that the problem has happened with clients	doi	29
Gender	The gender of the client	gen	150
Location	The place where the service is conducted	loc	477
Price	The price of the object	pri	32
Type	The type of the intent	typ	474

TABLE II
THE LIST OF 13 LABELS FOR THE REAL ESTATE DOMAIN

Label name	Descriptions	Label	#
Intent	Intent keyword of the online text/user	int	835
Object	The object that the user's intent refers to	obj	1157
Acreage	The acreage of the object	acr	865
Bathroom number	The number of bathrooms that the objec contains	bathnum	111
Balcony direction	The direction of the objects balcony	bdir	21
Bedroom number	The number of bedrooms that the objec contains	bednum	146
Contact	The information of whom represents the the object's owner, such as the owner, the agency...	ctt	1257
Door direction	The direction of the objects door/facade	ddir	1157
Facade size	The size of the object's facade	face	329
Floor number	The number of floors the object contains	fnum	442
Floor position	The index number of the floor where the object locates	fpos	84
Location	The location where the object locates	loc	1101
Price	The price of the object	prc	645

2) Pre–trained Embedding: In this model, we use pre–trained word embeddings to initialize our lookup table. We observe significant improvements using pre–trained word embeddings over randomly initialized ones. Embeddings are pre–trained using skip–gram, where each word is represented as a bag of character n–grams. A vector representation is associated to each n–gram character, words being represented as the sum of these representations. It is noted that different vectors are assigned to a word and an n-gram sharing the same sequence of characters. This method was presented by Bojanowski et al. (2016) [1] and published by Facebook FastText. After embedding, the output text file will contain the word vectors, one per line. Each line of the pre–trained lookup table starts with one word and follows with a 100-dimensional vector of that word. In our experiment the dataset for building pre-trained embeddings is the raw data that is not tagged. For **real estate domain**, the data set includes 1041 different Vietnamese words. For the **cosmetic** & **beauty** domain, it includes 1500 different Vietnamese words.

3) Dropout Training: When a large neural network is trained on a small training data set, it may perform poorly on held–out test data. This is called overfitting. In our experiments, we used dropout training [7] to reduce this problem. This can be done by randomly omitting some of the feature detectors on each training case. The term dropout refers to dropping out units (hidden and visible) in a neural network. By dropping a unit out, we temporarily remove it from the network, along with all its incoming and outgoing connections. The choice of which units to be dropped is random. In our experiment, we set the dropout rate to 0.3.

Table III
N-GRAM FEATURE TEMPLATES TO TRAIN THE CRFS MODEL

N–grams	Context predicate templates
1–grams	$[w_{-2}]$, $[w_{-1}]$, $[w_0]$, $[w_1]$, $[w_2]$
2–grams	$[w_{-2}w_{-1}]$, $[w_{-1}w_0]$, $[w_0w_1]$, $[w_1w_2]$
3–grams	$[w_{-2}w_{-1}w_0]$, $[w_{-1}w_0w_1]$, $[w_0w_1w_2]$

D. Feature Template for CRFs

One of the most important factors to improve the accuracy of extraction models is identifying appropriate feature templates. In our experiments, we utilized three type of feature templates, they are n–gram, look–up dictionary and regular expressions. Table III shows the first one, n–gram feature. We used 1–grams, 2–grams , and 3–grams. When combining consecutive tokens to form n–grams, we did not join two consecutive tokens if there is a punctuation mark between them. To collect state features we used the sliding window of size 5 to scan context predicates from data. To build our two dictionaries for look–up features, we look carefully at our two data sets to select list of brands, list of location, such as "*victoria secret*", "*wonjin*", "*Phan Chu Trinh*", "*Thanh Luong*". Some of consecutive tokens were joined and looked up in the dictionary to generate more context predicates. Moreover, we created regular expressions to improve the ability of predicting some special labels such as *price*, *acreage*, etc. We found that

Table IV
THE ACCURACY OF THE COSMETIC & BEAUTY DOMAIN EXPERIMENTS

	Precision	Recall	F1
LSTM-CRF (Char)	90.99%	87.19%	89.01%
LSTM-CRF (Char+Drop)	92.08%	89.37%	90.71%
LSTM-CRF (Char+Pre)	90.14%	89.25%	89.69%
LSTM-CRF (Char+Pre+Drop)	92.79%	89.60%	**91.17%**
CRFs	92.15%	73.49%	81.76%

Table V
THE ACCURACY OF THE REAL-ESTATE DOMAIN EXPERIMENTS

	Precision	Recall	F1
LSTM-CRF (Char)	91.94%	90.83%	**91.37%**
LSTM-CRF (Char+Drop)	90.39%	89.37%	89.87%
LSTM-CRF (Char+Pre)	89.98%	89.90%	89.94%
LSTM-CRF (Char+Pre+Drop)	90.23%	89.00%	89.53%
CRFs	87.21%	85.68%	86.43%

the look–up dictionary and regular expression features are very useful.

VI. EXPERIMENTAL RESULTS AND ANALYSIS

We built 5 models to conduct our experiments. The first one is the LSTM–CRF with character based model of word (Char). The second is the LSTM–CRF with Char and dropout training option (Drop). The third is the LSTM–CRF with Char and pre–trained embedding option (Pre). The fourth is LSTM–CRF with the combination of all three options Char+Pre+Drop. The last one is the CRFs model. For LSTM model, we used the tool Tagger (https://github.com/glample/tagger) and stack-LSTM-CRF (https://github.com/clab/stack-lstm-ner) with some modification to train. And we used the tool FlexCRFs published by Xuan-Hieu Phan [22] to conduct the CRF model experiments. For each model, we carefully trained and performed 5–fold cross–validation test. The experimental results for the two domains are described in Figure 5 and Figure 6. It shows that the results are quite stable over the 5 folds for both domains. It means that these 5 models can work well on these data sets. And in all 5 folds, the LSTM–CRF models always perform better than the CRFs model. The average accuracy of 5-fold for **cosmetic** & **beauty** domain and **real estate** domain are shown respectively in Table IV and Table V. For **cosmetic** & **beauty** domain, the LSTM-CRF (Char+Pre+Drop) achieved best accuracy. For **real estate** domain the LSTM-CRF (Char) is most accurate. This difference can caused by the distinct character in each data set. In general, the accuracy are remarkably high for all 5 models. To make it more clearly, we show the average over 5 folds of F_1–score for each label in two domains in the Figure 3 and Figure 4. The label <doi> in **cosmetic** & **beauty** domain and the label <bdir> in the **real estate** domain had lowest accuracy. This can be explained by their less frequency in the data sets. The label <loc> in the **real estate** domain had low accuracy even it appears frequently in the data, more than 400 times. The reason is the location in **real estate** domain is usually long and complicated.

VII. CONCLUSION

This paper presents our approach to the problem of intent parsing and extraction. To solve this problem, we have formulated this as a sequential segmentation. This is actually an information extraction from texts where we need to tag and identify intent information such as intent name, intent object, and other intent–related properties or constraints. We have also proposed the use of two advanced machine learning techniques, CRFs and LSTM, to build extraction models. These learning techniques have been proven to be suitable for dealing with labeling and segmenting sequence data. To evaluate our methods, we have prepared two annotated datasets for the two domains: **real estate** and **cosmetic** & **beauty**. The tag/label sets for the two domain were also carefully designed. We carefully conducted empirical studies on the two dataset and analyzed the results. The experimental results show that we can solve intent parsing and extraction with a significantly high accuracy and our information extraction and machine learning based approach is appropriate.

ACKNOWLEDGMENT

This work was supported by the project QG.15.29 from Vietnam National University, Hanoi (VNU).

REFERENCES

[1] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. *Enriching word vectors with subword information*, arXiv:1607.04606, 2016.
[2] Z. Chen, B. Liu, M. Hsu, M. Castellanos, and R. Ghosh. *Identifying intention posts in discussion forums*, HLT-NAACL. 2013.
[3] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu and P. Kuksa. *Natural language processing (almost) from scratch*, JMLR, 12:2493–2537, 2011.
[4] A. Graves, and S. Jrgen. *Framewise phoneme classification with bidirectional LSTM and other neural network architectures*, Neural Networks 18(5):602–610, 2005.
[5] V. Gupta, D. Varshney, H. Jhamtani, D. Kedia, S. Karwa. *Identifying purchase intent from social posts*, In Proc. of ICWSM, 2014.
[6] H.B. Hashemi, A. A.siaee, R. Kraft. *Query intent detection using convolutional neural networks*, In Proc. of WSDM QRUMS Workshop, 2016.
[7] G.E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever and R.R. Salakhutdinov. *Improving neural networks by preventing co–adaptation of feature detectors*, arXiv:1207.0580, 2012.
[8] S. Hochreiter, and S. Jrgen. *Long short-term memory*, Neural computation 9(8):1735–1780, 1997.
[9] B. Hollerit, M. Krll, M. Strohmaier. *Towards linking buyers and sellers: detecting commercial intent on twitter*, In Proc. of WWW, 2013.
[10] Z. Huang, X. Wei, and Y. Kai. *Bidirectional LSTM-CRF models for sequence tagging*, arXiv:1508.01991, 2015.
[11] M. Hwang, D.H. Jeong, J. Kim, S. Song, and H. Jung. *Activity inference for constructing user intention model*, Computer Science and Information Systems, 10(2):767–778, 2013.
[12] C.K.C. Jackie, X. Li *Sequence clustering and labeling for unsupervised query intent discovery*, In Proc. of WSDM, 2012.
[13] J.K. Kim, G. Tur, A. Celikyilmaz, B. Cao, Y.Y. Wang. *Intent detection using semantically enriched word embeddings*, In Proc. of SLT Workshop, IEEE, 2016.
[14] J. Lafferty, M. Andrew, and P. Fernando. *Conditional random fields: probabilistic models for segmenting and labeling sequence data*, In Proc. of ICML, 2001.
[15] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, C. Dyer. *Neural architectures for named entity recognition*, arXiv:1603.01360, 2016.
[16] Th.L. Luong, Th.H. Tran, Qu.T. Truong, Th.M.Ng. Truong, Th.Th. Phi, X.H. Phan. *Learning to filter user explicit intents in online vietnamese social media texts*, In Proc. of ACIID, 2016.

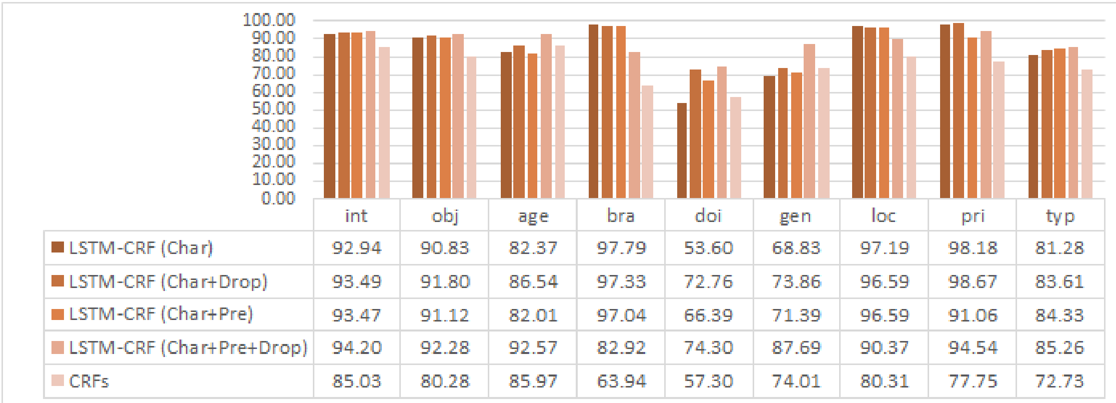

Figure 3. The average of F1-score for each label in cosmetic & beauty domain

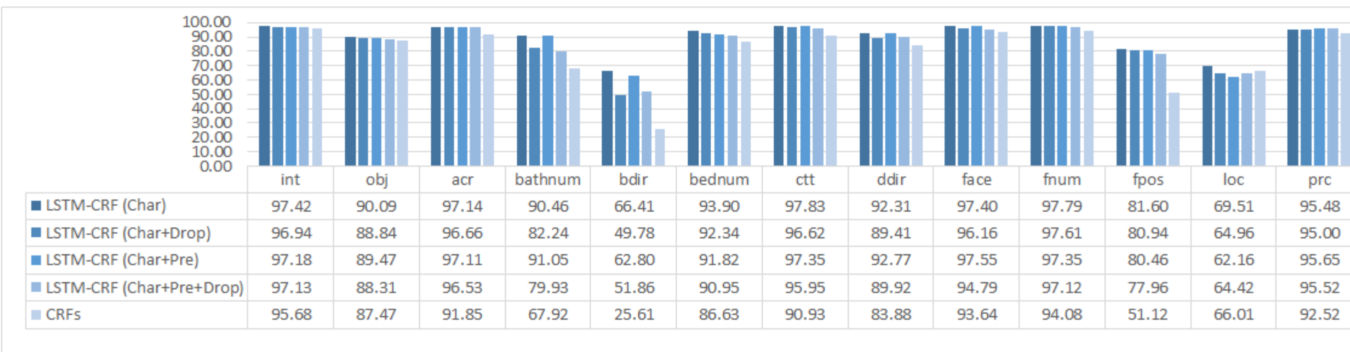

Figure 4. The average of F1-score for each label in real-estate domain

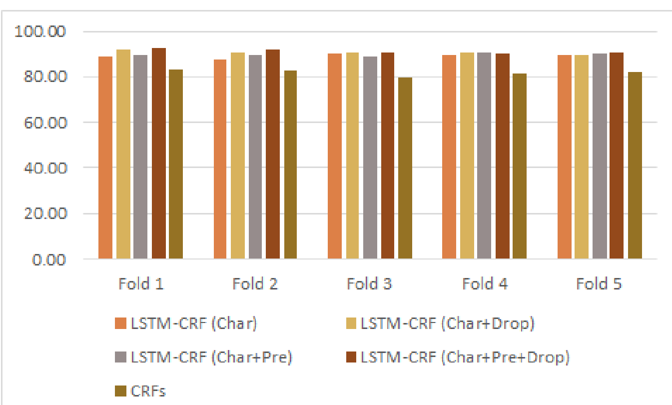

Figure 5. The accuracy of the 5-fold CV tests for cosmetic & beauty domain

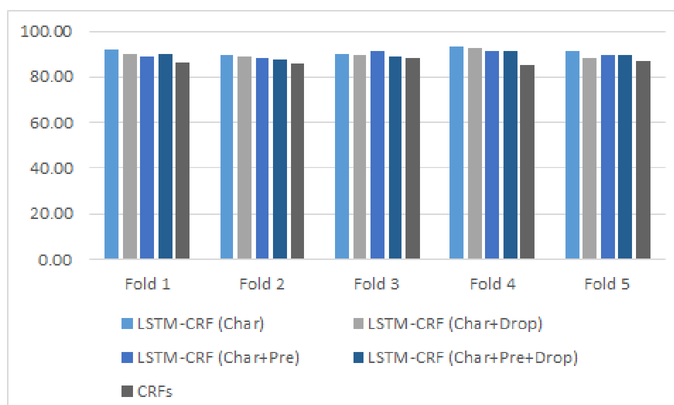

Figure 6. The accuracy of the 5-fold CV tests for real-estate domain

[17] Th.L. Luong, Qu.T. Truong, H.Tr. Dang, X.H. Phan. *Domain identification for intention posts on online social media*, In Proc. SoICT, 2016.

[18] X. Li. *Understanding the semantic structure of noun phrase queries*, In Proc. of ACL, 2010.

[19] X. Li, W. Ye-Yi, and A. Alex. *Learning query intent from regularized click graphs*, in Proc. of SIGIR, 2008.

[20] W. Ling, T. Luis, L. Marujo, R.F. Astudillo, S. Amir, C. Dyer, A.W. Black, and I. Trancoso. *Finding function in form: compositional character models for open vocabulary word representation*, arXiv:1508.02096, 2015.

[21] G.H. Nobari, T.S. Chua. *User intent identification from online discussions using a joint aspect-action topic model*, In Proc.of AAAI, 2014.

[22] X.H. Phan, L.M. Nguyen, C.T. Nguyen. *Flexible conditional random fields*, http://flexcrfs.sourceforge.net, 2004.

[23] X. Ren, Y. Wang, X. Yu, J. Yan, Z. Chen, J. Han. *Heterogeneous graph-based intent learning with queries, web pages and wikipedia concepts*, In Proc. of ICWSDM, 2014.

[24] J. Wang, G. Cong, W.X. Zhao, X. Li. *Mining user intents in Twitter: a semi-supervised approach to inferring intent categories for tweets*, In Proc. of AAAI, 2015.

[25] J. Xu, Q. Zhang, X. Huang. *Understanding the semantic intent of domain-specific natural language query*, In Proc. of IJCNLP, 2013.