ISSN 2007-9737

Mining Purchase Intent in Twitter

Rejwanul Haque, Arvind Ramadurai, Mohammed Hasanuzzaman, Andy Way

Dublin City University, School of Computing,
ADAPT Centre, Dublin,
Ireland

{firstname.lastname}@adaptcentre.ie

Abstract. Most social media platforms allow users to freely express their beliefs, opinions, thoughts, and intents. Twitter is one of the most popular social media platforms where users' post their intent to purchase. A purchase intent can be defined as measurement of the probability that a consumer will purchase a product or service in future. Identification of purchase intent in Twitter sphere is of utmost interest as it is one of the most long-standing and widely used measures in marketing research. In this paper, we present a supervised learning strategy to identify users' purchase intent from the language they use in Twitter. Recurrent Neural Networks (RNNs), in particular with Long Short-Term Memory (LSTM) hidden units, are powerful and increasingly popular models for text classification. They effectively encode sequences with varying length and capture long range dependencies. We present the first study to apply LSTM for purchase intent identification task. We train the LSTM network on semi-automatically created dataset. Our model achieves competent classification accuracy $(F_1 = 83\%)$ over a gold-standard dataset. Further, we demonstrate the efficacy of the LSTM network by comparing its performance with different classical classification algorithms taking this purchase intent identification task into account.

Keywords. Social media, purchase intent, mining, user generated content.

1 Introduction

Sharing personal thoughts, beliefs, opinions, and intents on the internet (especially on social media platforms) has become an essential part of life for millions of users all around the world. Twitter[1], one of the popular social media platforms, where users put forth their intent to purchase products or services and looks for suggestions that could assist them. To exemplify, 'I wanna buy an iPhone this week!' indicates the user's intent for buying an Apple iPhone soon. Essentially, identification and classification of such user generated contents (UGC) have twofold benefits: (a) commercial companies could exploit this to build their marketing tool/strategy, and (b) it could benefit social media users with the suggestions of the products or services that they want to purchase.

Gupta et al. [7] investigated the relationship between users' purchase intent from their social media forums such as Quora[2] and Yahoo! Answers[3]. They mainly carried out text analysis (e.g. extracting features, such as purchase action words, using the dependency structure of sentences) to detect purchase intent from UGC. In another study [18], the authors investigated the problem of identifying purchase intent. In particular, the authors (i.e. [18]) proposed a graph-based learning approach to identify intent tweets and classify them into six categories, namely 'Food & Drink', 'Travel', 'Career & Education', 'Goods & Services', 'Event & Activities' and 'Trifle'. For this, they retrieved tweets with a bootstrap method, with using a list of seed intent-indicators (e.g. 'want to'), and manually created training examples from the collected tweets. There is a potential problem in their data set since it was created based on a handful of keywords (i.e. indent-indicators).

[1] https://twitter.com/

[2] www.quora.com

[3] www.answers.yahoo.com

In reality, there could have many lexical variations of an intent-indicator. For example, any of these following intent-indicators can take the place of 'want to': 'like to', 'wish to', and 'need to'. Tweets often include misspelled short or long words depending on user's emotions, thoughts and state of mind.

For example, when a user is really excited to buy a car soon, his purchase intent tweet can be 'I liiiiiiiiiike to buy the SUV this month!!!' that includes an intent-indicator 'liiiiiiiiike to' which has a misspelled long word, 'liiiiiiiiiike'.

In this work, in order to capture new tweets that are good paradigms of purchase intentions, we adopted a seed intent-indicators expansion strategy using a query expansion technique [14]. This technique has essentially helped to increase the coverage of keywords in our training data.

We manually create a labeled training data with the tweets that were extracted using a python API, given the expanded seed list of the intent-indicators. In order to identify users' purchase intention in tweets, we present a RNN model [17, 20] with LSTM units [8] (cf. Section 6). To summarize, our main contributions in this paper are as follows:

1. We are the first to apply the deep learning techniques for the users' purchase intent identification task in social media platform.
2. We create a gold-standard training data set, which, in practice, can be viewed as an ideal data set for the users' purchase intent identification task in Twitter.

The remainder of the paper is organised as follows. In Section 2, we discuss related work. Section 3 presents an existing training data that was previously used in the purchase intent identification task. In Section 4, we detail how we created training data for our experiments. In Section 5, we present our experimental methodology. Section 6 presents our experimental set-up, results and analysis, while Section 7 concludes, and provides avenues for further work.

2 Related Work

Identifying wishes from texts [16, 6] is apparently a new arena in natural language processing (NLP). Notably, Ramanand et al. [16] focus on identifying wishes from product reviews or customer surveys, e.g. a desire to buy a product. They primarily describe linguistic rules that can help detect these 'wishes' from text. In general, their rule-based method for identifying wishes from text proved to be effective. However, the creation of rules is a time-consuming task, and their coverage is not satisfactory. Detection of users' purchase intent in social media platform is close to the task of identifying wishes in product reviews or customer surveys.

In information retrieval (IR), query intent can broadly be classified into two categories: query type [10, 3] and user type [2, 13, 9]. The focus on this paper is to identify and classify tweets that explicitly express users' purchase intents. In this sense, this work can be kept under of the first category. To the best of our knowledge, the most relevant works to ours come from [7, 18]. In fact, to a certain extent, our proposed methods can be viewed as the extension of [18].

[7] investigated the problem of identifying purchase intent in UGC, with carrying out an analysis of the structure and content of posts and extracting features from them. They make use of the linguistic preprocessors for feature extraction, such as dependency parser and named entity recogniser, which are only available for a handful of languages.

[18] presented a graph-based learning approach to inferring intent categories for tweets. [18] primarily focus on identifying and classifying tweets that explicitly express user's purchase intentions. In order to prepare a training data with tweets that express user's purchase intentions, [18] proposed a bootstrap-based extraction model that made use of a seed list of purchase-indicators (e.g. 'want to'). They took help of manual annotators to classify the collected tweets into six different intent categories. The major disadvantage of these methods [7, 18] lies with their data set since their training data is based on a handful of keywords. In our work, we encountered this problem with employing an query

expansion technique [14], which has essentially helped to increase the coverage of keywords in our training data. We are the first to train our models with deep learning technique (RNN with LSTM hidden units) for this problem, i.e. purchase intent identification task in social media platform.

3 Existing Dataset

This section details an existing labeled training data in which each tweet is associated with an appropriate purchase intent or non-intent category. The creation of this data set was based on a limited set of keywords. A brief overview of this dataset is provided below.

As mentioned in Section 2, [18] applied a bootstrapping based method to retrieve intent tweets from Twitter, given a seed set of intent-indicators, (e.g. 'want to'). A manual annotation process was carried out on those extracted tweets that contain at least one intent-indicator. In short, tweets were distinguished as intent and non-intent tweets and the intent tweets were categorised into six different categories, namely 'Food and Drink', 'Travel', 'Education and Career', 'Goods and Services', 'Event and Activities' and 'Trifle'. [18] shared their training data with us. From now, we call this data set *Dataset1*. The statistics of Dataset1 can be found in [18], which we also report in Table 1. As can be seen from Table 1, Dataset1 consists of 2,263 labeled tweets, with six intent and non-intent categories.

Table 1. The statistics of the existing training data set, Dataset1

Category	tweets	%
Food & Drink	245	11.50%
Travel	187	8.78%
Carrer & Education	159	7.46%
Goods & Services	251	11.78%
Event & Activities	321	15.07%
Trifle	436	20.47%
Non-intent	531	24.92%
Total	2,263	

4 Our Dataset

This section details creation of a new training data. First, we explain why the existing data (i.e. Dataset1), to a certain extent, is inadequate for this task. Then, we demonstrate how we created a new dataset. The created dataset, in practice, is to be an ideal data set for addressing this problem.

4.1 Variation of Intent-Indicators

The expression of interest of a Twitter user may be associated with the user's state of mind, emotion, or other phenomenon. Hence, the different Twitter users can express their thoughts of interest in numerous ways. For example, the users may show theirs interest to purchase a product with any of the following intent-indicators: 'want to','need to','like to','wish to', and 'hope to'. Spelling mistake is a common phenomenon in tweets (e.g. short form, noisy long form). Hence, tweets may include misspelled intent-indicators.

For example, when a user is really excited to buy a product soon, his purchase intent tweet can be 'I wannnnttt to buy an iPhone!!!!!!'. Similarly, intent indicators can be specified as 'n33d to', 'h0pe to', 'wnt to' and so on. All the prior studies in this direction do not take into the consideration of different ways by which an intent indicator can be specified. In this work, we aim to make the list of purchase intent indicators as exhaustive as possible, with taking the nature of the user generated contents in this media into consideration. In the next section we describe how we expand the existing list of seed purchase intend indicators.

4.2 Expanding the List of Intent-Indicators

As discussed above, the existing data set (i.e. Dataset1) has limited coverage of the indent-indicators. In order to increase the coverage, and to capture new tweets that are good paradigms of purchase intentions, we expand the list of intent-indicators[4] using a query expansion technique. This is accomplished with a continuous distributed vector representation of words using

[4]We obtained the initial list of intent-indicators from [18]

ISSN 2007-9737

the continuous Skip-gram model (also known as Word2Vec) proposed by [14], by maximizing the objective function:

$$\frac{1}{|V|}\sum_{n=1}^{|v|}\sum_{-c\leq j\leq c, j\neq 0} logp(w_{n+1}|w_n), \qquad (1)$$

where $|V|$ is the size of the vocabulary in the training set and c is the size of context window. The probability $p(w_{n+j}|w_n)$ is approximated using the hierarchical softmax.

Table 2. Top 10 similar words/phrases of the seed intend indicators: 'want', 'need', 'wish' and 'like'

	Intent-indicators			
Top-10 similar words/phrases	want	need	wish	like
	wamt	neeed	Wish	lile
	wan't	nees	wished	likr
	wanr	neeeed	wishh	llike
	want/need	neeeeed	whish	likw
	wabt	meed	Wishhhh	lik
	wNt	Need	Wished	lkke
	wnat	n99ed	Wishin	like
	/want/	neex	WISH	lije
	need/want	need/want	lwish	lke
	eant	neeeeeed	wishhh	lyk

In our case, we used a pre-trained 200-dimensional GolVe vectors [15]. Given the vector representations for the intend-indicators, we calculate the similarity scores between words/phrase pairs in the vocabulary using cosine similarity. Top 10 similar words/phrase for each seed intent-indicator in our list are selected for the expansion of initial seed list. For an example, in Table 2, we list the top 10 similar intent-indicators of four seed intent-indicator: 'want', 'need', 'wish' and 'like'. Finally, in order to weed out irrelevant intent-indicators, if any, from the list of the expanded intent-indicators, a manual inspection was carried out.

4.3 Collecting Tweets with the Expanded seed Intent-Indicators

We extract tweets using a Pyhton library, Tweepy,[5] from Twitter. For extraction, we use the expanded list of seed intent-indicators (cf. Section 4.2). Potentially, each of the extracted tweets contains at least one intent-indicator. In Table 3, we show a few of those tweets that were collected from Twitter given the seed intent-indicator: 'n33d'.

Table 3. A few of the tweets collected with the seed intent-indicator: 'n33d'

tweets with intent-indicator: 'n33d'
I n33d yall to be more active on younow plz and thanks
I n33d more youtubers to watch
I n33d to make some fucking music
I n33d to inhale these pizz@ pringle
I am so hungry people got to watch out n33d foOOoOod
I n33d to stop buying jackets
I was at ritz last friday shown out y3w n33d to go out to da club wit m3
I think I need to go get some therapy

4.4 Labeling Collected Purchase Intent Tweets

We randomly sampled a set of 2,500 tweets from the list of the collected tweets that contain at least one intent-indicator, and another set of 2,500 tweets from Twitter, each of them contains no intent-indicators. During sampling we ensure that none of the tweets overlaps with those from Dataset1 (cf. Section 3). Then, we applied a noise cleaning method on the tweets, i.e. all the null values, special characters, hashtags were removed from tweets. This cleaning process was carried out with a manual editor who has excellent English skills and good knowledge on tweets. After manual cleaning, we get a set of 4,732 tweets.

[5] http://docs.tweepy.org/en/v3.5.0/api.html

As mentioned earlier in the paper, [18] defined a set of purchase intent categories (six) in order to classify those tweets that express users' purchase intention. Following [18] we label each of the collected clean tweets with either one of the purchase intent categories or the non-intent category. The manual annotation process is accomplished with a GUI that randomly displays a tweet from the set of 4,732 tweets. The GUI lists the six intent (i.e. 'Food and Drink', 'Travel', 'Education and Career', 'Goods and Services', 'Event and Activities' and 'Trifle') categories and the sole non-intent category as in [18]. For the annotation purposes we hired three annotators who are native English speakers and have excellent knoweldge on UGCs (i.e. tweets). The annotators are instructed to follow the following rules for labeling a tweet:

— label each tweet with an appropriate category listed on GUI,

— skip a tweet for annotation if you are unsure about user's purchase intention in the tweet or its intention category,

— skip those tweets for annotation that are unclear and includes characters of other languages or noisy characters,

— label those tweets with the non-intent category that express negative intents (e.g., 'don't want to').

On completion of the annotation task, we obtained 4,210 tweets, each of which is associated with at least one tag[6]. Since we have three annotators and three values are associated with the most of tweets of the set of 4,210 labeled tweets, final class for a tweet is determined with two out of three voting logic.

Thus, 635 annotated tweets were not considered in the final annotated set due to the disagreements of all three annotators. The final set of annotated tweets contains 3,575 entries. From now, we call this data set *Dataset2*, whose statistics are reported in Table 4.

[6]At least, one out of three manual annotators label each of the 4,210 tweets.

Table 4. The statistics of the new training data set, Dataset2

Category	tweets	%
Food & Drink	285	8.0%
Travel	214	6.0%
Carrer & Education	164	4.6%
Goods & Services	387	10.8%
Event & Activities	344	9.6%
Trifle	450	12.6%
Non-intent	1,803	50.4%
Total	3,575	

On completion of the annotation process, inter-annotator agreement was computed using Cohen's kappa [4] at tweet level. For each tweet we count an agreement whenever two out three annotators agree with the annotation result. We found the kappa coefficient to be very high (i.e. 0.64) for the annotation task. This indicates that our tweet labeling task is to be excellent in quality.

Table 5. The statistics of the combined training data set, ComDataset

Category	tweets	%
Food & Drink	530	9.1%
Travel	401	6.9%
Carrer & Education	323	5.5%
Goods & Services	538	9.2%
Event & Activities	665	11.4%
Trifle	886	15.2%
Non-intent	2,336	40.0%
Total	5,838	

4.5 Combined Training Data

For our experiments we merged the training examples of Dataset1 and Dataset2. From now, we call the combined training set *ComDataset*. The statistics of ComDataset are reported in Table

5. For our experiments we randomly selected 1,000 examples from ComDataset, and create a test set with 500 examples and a validation set with 500 examples. The set of remaining 4,838 examples from ComDataset was considered as the training set.

5 Methodology

5.1 LSTM Network

Nowadays, RNN, in particular with LSTM [8] hidden units, has been proved to be an effective model for many classification tasks in NLP, e.g. sentiment analysis [19], text classification [11, 21]. RNN is an extension of the feed-forward NN, which has the gradient vanishing or exploding problems. LSTM deals with the exploding and vanishing gradient problems of RNN. An RNN composed of LSTM hidden units is often called an LSTM network. A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. More formally, each cell in LSTM can be computed as follows:

$$X = \begin{bmatrix} h_{t-1} \\ x_t, \end{bmatrix} \tag{2}$$

$$f_t = \sigma(W_f \cdot X + b_f), \tag{3}$$

$$i_t = \sigma(W_i \cdot X + b_i), \tag{4}$$

$$o_t = \sigma(W_o \cdot X + b_o), \tag{5}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot tanh\ (W_c \cdot X + b_c), \tag{6}$$

$$h_t = o_t \odot tanh\ (c_t), \tag{7}$$

where $W_i, W_f, W_o \in \mathbb{R}^{d \times 2d}$ are the weighted matrices and $b_i, b_f, b_o \in \mathbb{R}^d$ are biases of LSTM, which need to be learned during training, parameterising the transformations of the input, forget and output gates, respectively. σ is the sigmoid function, and $\odot$ stands for element-wise multiplication. x_t includes the inputs of LSTM cell unit. The vector of hidden layer is h_t. The final hidden vector h_N represents the whole input tweet, which is passed to *softmax* layer after linearizing it into a vector whose length is equal to the number of class labels. In our work, the set of class labels includes intent and non-intent categories.

5.2 Classical Supervised Classification Models

Furthermore, we compare the deep learning model with the classical classification models. We employ the following classical supervised classification techniques:

— Baseline 1: Logistic Regression (LR),

— Baseline 2: Decision Tree (DT),

— Baseline 3: Random Forest (RF),

— Baseline 4: Naïve Bayes (NB).

These classical learning models (LR, DT, RF and NB) can be viewed as the baselines in this task. Thus, we obtain a comparative overview on the performances of different supervised classification models including LSTM network. Note that we consider default set-ups of an well-known machine learning library for our baseline classifiers (cf. Section 6).

6 Experiments

This section details the building of different classification models. In order to build LR, DT, RF and NB classification models, we use the well-known scikit-learn machine learning library,[7] and performed all the experiments with default parameters set by scikit-learn. As for the representation space, each tweet was represented as a vector of word unigrams weighted by their frequency in the tweet. For building our neural network (NN) and training the model we use Lasagne library[8].

Our RNN model includes LSTM units. The size of input layer of the NN is 12,000. We employ layer normalisation [1] in the model. Dropout [5] between layers is set to 0.10. The size of embedding and hidden layers are 512 and 1024. The models are trained with Adam optimizer [12], with learning-rate set to 0.0003 and reshuffling the training corpora for each epoch. We use the learning rate warm-up strategy for Adam. The validation on development set is performed using cross-entropy cost function.

Table 6. Accuracy of classification models (set-up 1: intent and non-intent) measured with precision, recall and F_1-score metrics

		P	R	F_1
LR	Intent	0.79	0.89	0.82
	Non-intent	0.88	0.73	0.80
	avg/total	0.82	0.81	0.81
DT	Intent	0.75	0.81	0.78
	Non-intent	0.80	0.74	0.77
	avg/total	0.78	0.77	0.77
RF	Intent	0.76	0.89	0.82
	Non-intent	0.87	0.73	0.79
	avg/total	0.82	0.81	0.81
NB	Intent	0.76	0.89	0.82
	Non-intent	0.88	0.73	0.80
	avg/total	0.82	0.81	0.81
RNN	Intent	0.82	0.86	0.84
	Non-intent	0.88	0.77	0.83
	avg/total	0.85	0.82	**0.83**

The RNN model is trained up-to 20 epochs, and we set mini-batches of size 32 for update.

We observe the learning curve of the classification models with the following experimental set-up:

— Set-up 1: classifying tweets into the two classes: intent and non-intent. In this case, all intent sub-classes are merged into a one single intent class (cf. Table 5).

— Set-up 2: classifying tweets into seven classes: six intent categories ('Food and Drink', 'Travel', 'Education and Career', 'Goods and Services', 'Event and Activities' and 'Trifle') and one non-intent category.

— Set-up 3 (one vs all): in this set-up we select a particular intent class, and the remaining intent sub-classes are merged into one single class. Like the first set-up (Set-up 1), this one is a binary classification task. In order to test classifiers in this set-up, we chose the following two intent classes: 'Goods and Services' and 'Trifle'.

[7] https://scikit-learn.org/stable/

[8] https://lasagne.readthedocs.io/en/latest/

6.1 Results and Discussion

We evaluate the performance our classifiers and report the evaluation results in this section. In order to measure classifier's accuracy on the test set, we use three widely-used evaluation metrics: precision, recall and F_1 measures. Note that we could not directly compare the approach of [18] with ours since the source code of their model is not freely available to use. We report the evaluation results on our gold standard test set obtained with the experimental set-up 'Set-up 1' in Table 6. Here, we draw a number of observations from the evaluation results presented in Table 6:

1. We see excellent performance with all classifiers for both intent and non-intent catergories.
2. As can be seen from Table 6, the precision scores are slightly higher than the recall scores for the non-intent category. The opposite scenario is observed with the intent category. When we compare intent and non-intent categories in terms of precision and recall, we see differences of the recall scores are higher than that of the precision scores irrespective of the classification models.
3. Irrespective of the classification models, the accuracy (F_1) of identifying purchase intent tweets is slightly better than that of identifying non-intent tweets.
4. When we compare the scores of different classification models on the test set, we see that the RNN model becomes the winner, with achieving a F_1 of 0.83 on the gold-standard test set.

Next, we report the evaluation results obtained with the experimental set-up 'Set-up 2' (cf. Section 6) in Table 7 and 8. This time, the classification task involves seven output classes, i.e. six intent classes and the sole non-intent class.

Table 7. Accuracy of the LR, DT and RF models (set-up 2) on six intent classes and one non-intent class measured with precision, recall and F_1-score metrics.

		P	R	F_1
LR	Food and Drink	0.87	0.83	0.85
	Travel	0.69	0.57	0.62
	Education & Career	1.0	0.51	0.68
	Goods & Services	0.77	0.56	0.65
	Event & Activities	0.71	0.51	0.68
	Trifle	0.47	0.45	0.46
	Non-intent	0.78	0.94	0.86
	avg/total	0.75	0.75	0.74
DT	Food and Drink	0.54	0.42	0.30
	Travel	0.34	0.37	0.36
	Education & Career	0.54	0.54	0.54
	Goods & Services	0.60	0.57	0.59
	Event & Activities	0.29	0.31	0.30
	Trifle	0.35	0.47	0.40
	Non-intent	0.80	0.76	0.78
	avg/total	0.63	0.61	0.62
RF	Food and Drink	0.61	0.69	0.65
	Travel	0.49	0.54	0.51
	Education & Career	0.71	0.57	0.63
	Goods & Services	0.62	0.71	0.66
	Event & Activities	0.54	0.28	0.37
	Trifle	0.35	0.34	0.34
	Non-intent	0.80	0.84	0.82
	avg/total	0.67	0.68	0.67

Table 8. Accuracy of the NB and RNN models (set-up 2) on six intent classes and one non-intent class measured with precision, recall and F_1-score metrics.

		P	R	F_1
NB	Food & Drink	1.0	0.12	0.22
	Travel	1.0	0.03	0.06
	Education & Career	1.0	0.06	0.11
	Goods & Services	0.89	0.18	0.30
	Event & Activities	1.0	0.03	0.05
	Trifle	0.43	0.04	0.07
	Non-intent	0.54	1.00	0.70
	avg/total	0.69	0.55	0.43
RRN	Food & Drink	0.90	0.85	0.88
	Travel	0.76	0.68	0.72
	Education & Career	0.74	0.73	0.74
	Goods & Services	0.86	0.67	0.75
	Event & Activities	0.92	0.96	0.94
	Trifle	0.57	0.54	0.55
	Non-intent	0.67	0.91	0.77
	avg/total	0.77	0.76	**0.76**

Note that due to the space constraints, we report the results in two tables (i.e. Tables 7 and 8). Here, we draw a number of observations from the evaluation results presented in Table 7 and 8:

1. In general, we get high precision and low recall scores for the intent categories. For the non-intent class, most of the cases, as in above, the scenario is the other way round.
2. As far as the scores obtained with the F_1 metric are concerned, we see that the RNN and LR classifiers performed reasonably, and the remaining classifiers (i.e. DT, NB and RF) performed moderately.
3. When we compare different classification models, we see, as in Set-up 1, the RNN model becomes the winner, with achieving F_1 of 0.76 (average) on the gold-standard test set. When we see F_1 scores for the intent and non-intent classes, we see that

the RNN classifiers performs consistently and outperforms all classical classification models in most of the cases.

4. As can be seen from Table 8, the recall scores of NB classifier are below par, and even in some cases, those are very poor. We recall Table 5 where we can see the presence of class imbalance in the training data. For instance, 6.9% and 5.5% training examples belong to 'Travel' and 'Career and Education' classes, respectively. This could be one of the reasons why classifiers performed poorly for some categories. This phenomenon is also corroborated by Gupta et al. [7] who built classifiers with a training data having the class imbalance issues.

Now, we observe the learning curve of the classifiers with the third experimental set-up (i.e. 'Set-up 3', cf. Section 6) where a particular intent category (e.g. 'Goods and Services' or 'Trifle') is held and the rest of the intent categories are merged into a single category. In this set-up, we remove those examples from the test and development sets that include the non-intent target class. Then, we test our classifiers on the resulting test set and obtain the evaluation results, which are reported in Table 9 ('Goods and Services') and in Table 10 ('Trifle'). Here, we draw a number of observations from the evaluation results presented in Table 9 and 10:

1. We see an excellent performance across the classifiers for 'Goods and Services' and the combined intent category.
2. We get a mix bag of results across the classifiers and metrics for 'Trifle' and the combined intent category .
3. Like the above experimental set-ups, in this set-up, the RNN models proved to be superior than the other classical classification models in identifying users' purchase intent type in tweets. The RNN model produces an accuracy of F_1 of 0.95 (average) on the test set when 'Goods and Services' category is considered. As far as the 'Trifle' category is concerned, the RNN model gives an accuracy of F_1 (average) of 0.83 on the test set.

Table 9. Accuracy of classification models (set-up 3: 'Goods and Services' and a combined class from the rest of the intent categories measured with precision, recall and F1 metrics

		P	R	F_1
LR	Goods & Services	0.88	0.76	0.82
	Intent	0.90	0.96	0.93
	avg/total	0.90	0.90	0.90
DT	Goods & Services	0.80	0.76	0.78
	Intent	0.90	0.92	0.91
	avg/total	0.87	0.87	0.87
RF	Goods & Services	0.80	0.72	0.76
	Intent	0.89	0.92	0.91
	avg/total	0.86	0.86	0.86
NB	Goods & Services	0.88	0.47	0.61
	Intent	0.81	0.97	0.89
	avg/total	0.83	0.82	0.80
RNN	Goods & Services	0.92	0.98	0.95
	Intent	0.94	0.98	0.96
	avg/total	0.93	0.98	**0.95**

4. When we consider the recall scores in Table 10, we see most of the classifiers performed below par with the 'Trifle' category. This anomaly needs to be investigated and we keep this topic as a subject of future work.

7 Conclusion

In this paper, we presented supervised learning models to identify users' purchase intent from the tweet data. We present the first study to apply LSTM network for purchase intent identification task. With our RNN classifiers we achieved

Table 10. Accuracy of classification models (set-up 3: 'Trifle' and a combined class from the rest of the intent categories measured with precision, recall and F1 metrics

		P	R	F_1
LR	Trifle	0.70	0.43	0.54
	Intent	0.83	0.94	0.88
	Avg/total	0.80	0.81	0.79
DT	Trifle	0.53	0.45	0.49
	Intent	0.82	0.86	0.84
	avg/total	0.75	0.76	0.75
RF	Trifle	0.66	0.38	0.48
	Intent	0.81	0.93	0.87
	avg/total	0.77	0.79	0.77
NB	Trifle	1.00	0.09	0.17
	Intent	0.76	1.00	0.87
	avg/total	0.82	0.77	0.69
RNN	Trifle	0.88	0.69	0.77
	Intent	0.93	0.87	0.89
	avg/total	0.91	0.78	**0.83**

competent accuracy (F_1 ranging from 0.76 to 0.95) in all classification tasks. This shows applicability of the deep learning algorithms to a classification task where a tiny training data is available.

Further, we demonstrated the efficacy of the LSTM network by comparing its performance with different classifiers. The major portion of the paper describes the way we created our own training data. The existing training data set for this task was not satisfactory as it is limited with a set of keywords. We semi-automatically created training data set, with employing a state-of-the-art query expansion technique [14]. This has essentially helped to increase the coverage of keywords in our training data.

In future, we intend to make our gold standard data set available to the NLP community. We also plan to test our method on different social media platform, e.g. Facebook,[9] and with different languages. We also intent to apply our methods to a cross-lingual social platform. We plan to increase the size of training examples for those classes for which we have lesser proportion of training examples. This could encounter the class imbalance problem in our training data.

[9]https://www.facebook.com/

Acknowledgements

The ADAPT Centre for Digital Content Technology is funded under the Science Foundation Ireland (SFI) Research Centres Programme (Grant No. 13/RC/2106) and is co-funded under the European Regional Development Fund. This project has partially received funding from the European Union's Horizon 2020 research and innovation programme under Marie Skłodowska-Curie Grant Agreement No. 713567, and the publication has emanated from research supported in part by a research grant from SFI under Grant Number 13/RC/2077.

References

1. **Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016).** Layer normalization. *CoRR*, Vol. abs/1607.06450.
2. **Beitzel, S. M., Jensen, E. C., Lewis, D. D., Chowdhury, A., & Frieder, O. (2007).** Automatic classification of web queries using very large unlabeled query logs. *ACM Transactions on Information Systems (TOIS)*, Vol. 25, No. 2, pp. 9.
3. **Cao, H., Hu, D. H., Shen, D., Jiang, D., Sun, J.-T., Chen, E., & Yang, Q. (2009).** Context-aware query classification. *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, ACM, Boston, MA, pp. 3–10.
4. **Cohen, J. (1960).** A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, Vol. 20, No. 1, pp. 37–46.
5. **Gal, Y. & Ghahramani, Z. (2016).** A theoretically grounded application of dropout in recurrent neural networks. *CoRR*, Vol. abs/1512.05287.
6. **Goldberg, A. B., Fillmore, N., Andrzejewski, D., Xu, Z., Gibson, B., & Zhu, X. (2009).** May all your wishes come true: A study of wishes and how to recognize them. *Proceedings of HLT-NAACL: Human Language Technologies: The 2009 Annual*

Conference of the North American Chapter of the Association for Computational Linguistics, Boulder, CO, pp. 263–271.

7. **Gupta, V., Varshney, D., Jhamtani, H., Kedia, D., & Karwa, S. (2014).** Identifying purchase intent from social posts. *Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media*, Ann Arbor, Michigan, pp. 180–186.
8. **Hochreiter, S. & Schmidhuber, J. (1997).** Long short-term memory. *Neural computation*, Vol. 9, No. 8, pp. 1735–1780.
9. **Hu, J., Wang, G., Lochovsky, F., Sun, J.-t., & Chen, Z. (2009).** Understanding user's query intent with wikipedia. *Proceedings of the 18th international conference on World wide web*, ACM, Madrid, Spain, pp. 471–480.
10. **Jansen, B. J., Booth, D. L., & Spink, A. (2008).** Determining the informational, navigational, and transactional intent of web queries. *Information Processing & Management*, Vol. 44, No. 3, pp. 1251–1266.
11. **Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016).** Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.
12. **Kingma, D. P. & Ba, J. (2014).** Adam: A method for stochastic optimization. *CoRR*, Vol. abs/1412.6980.
13. **Li, X., Wang, Y.-Y., & Acero, A. (2008).** Learning query intent from regularized click graphs. *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, New York, NY, pp. 339–346.
14. **Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013).** Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, pp. 3111–3119.
15. **Pennington, J., Socher, R., & Manning, C. (2014).** Glove: Global vectors for word representation. *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, Doha, Qatar, pp. 1532–1543.
16. **Ramanand, J., Bhavsar, K., & Pedanekar, N. (2010).** Wishful thinking: finding suggestions and'buy'wishes from product reviews. *Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text*, Los Angeles, CA, pp. 54–61.
17. **Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986).** Learning representations by back-propagating errors. *Nature*, Vol. 323, No. 6088, pp. 533.
18. **Wang, J., Cong, G., Zhao, W. X., & Li, X. (2015).** Mining user intents in twitter: A semi-supervised approach to inferring intent categories for tweets. *Twenty-Ninth AAAI Conference on Artificial Intelligence*, Austin, TX, pp. 318–324.
19. **Wang, Y., Huang, M., Zhao, L., et al. (2016).** Attention-based LSTM for aspect-level sentiment classification. *Proceedings of the 2016 conference on empirical methods in natural language processing*, Austin, TX, pp. 606–615.
20. **Werbos, P. J. (1990).** Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, Vol. 78, No. 10, pp. 1550–1560.
21. **Zhou, P., Qi, Z., Zheng, S., Xu, J., Bao, H., & Xu, B. (2016).** Text classification improved by integrating bidirectional lstm with two-dimensional max pooling. *arXiv preprint arXiv:1611.06639*.

Article received on 21/01/2019; accepted on 17/02/2019.
Corresponding author is Rejwanul Haque.