Reddit Temporal N-gram Corpus and its Applications on Paraphrase and Semantic Similarity in Social Media using a Topic-based Latent Semantic Analysis

Anh Dang[1], Abidalrahman Moh'd[1], Aminul Islam[2], Rosane Minghim[3], Michael Smit[4], and Evangelos Milios[1]

[1]{anh,amohd,eem}@cs.dal.ca, Dalhousie University, 6050 University Avenue, Halifax, NS, Canada B3H 4R2

[2]aminul@louisiana.edu, School of Computing and Informatics, University of Louisiana at Lafayette Lafayette, LA, USA 70503

[3]rminghim@icmc.usp.br, University of São Paulo-USP, ICMC, São Carlos, Brazil

[4]mike.smit@dal.ca, School of Information Management, Dalhousie University, 6100 University, Halifax, NS, Canada B3H 4R2

Abstract

This paper introduces a new large-scale n-gram corpus that is created specifically from social media text. Two distinguishing characteristics of this corpus are its monthly temporal attribute and that it is created from 1.65 billion comments of user-generated text in Reddit. The usefulness of this corpus is exemplified and evaluated by a novel Topic-based Latent Semantic Analysis (TLSA) algorithm. The experimental results show that unsupervised TLSA outperforms all the state-of-the-art unsupervised and semi-supervised methods in SEMEVAL 2015: paraphrase and semantic similarity in Twitter tasks.

1 Introduction

A word n-gram is a continuous sequence of n words from a corpus of texts or speech. Word n-gram language models are widely used in Natural Language Processing (NLP), such as speech recognition, machine translation, and information retrieval. The effectiveness of a word n-gram language model is highly dependent on the size and coverage of its training corpus (Clarke et al., 2002). A simple algorithm can outperform a more complicated algorithm if it uses a larger corpus (Norvig, 2008). Many large-scale corpora (Brants and Franz, 2006; Baroni et al., 2009; Wang et al., 2010b) based on web contents have been created for this purpose. As the use of social media is increasing, Online Social Networks (OSNs) have become a norm to spreading news, rumours, and social events (Kwak et al., 2010). This growing usage of social media has created both challenges and opportunities. One major challenge is that social media data is intrinsically short and noisy. A study by (Wang et al., 2010a) revealed that different text corpora have significantly different properties and lead to varying performance in many NLP applications. More importantly, we observe that there is no existing large-scale n-gram corpus that is created specifically from social media text. This has motivated us to create an n-gram corpus that is derived from 1.65 billion comments in the Reddit corpus (Baumgartner, July 2015) and make it available to the research community. There are two main features of this corpus that do not exist in the available large-scale corpora in the literature: monthly time-varying (temporal) and purely social media text. This corpus will allow researchers to analyze and make sense of massive social network text, such as finding corresponding terms across time (Zhang et al., 2015) and improving named entity recognition in tweets (Li and Liu, 2015). Moreover, a cloud-based visualization interface is implemented to allow end users to query any n-gram from the corpus.

Although there are many applications that can be derived from this corpus, in this paper, we use the Paraphrase Identification (PI) and Semantic Similarity (SS) tasks of SEMEVAL 2015 (Xu et al., 2015) to exemplify the usefulness of this corpus. Paraphrases are words, phrases or sentences that have the same meaning, but their vocabulary may be different (Xu et al., 2015). PI and SS tasks have a strong correlation, as both focus on the underlying structural and semantic similarity between two texts (e.g., "selfie" is a paraphrase of "picture of myself"). Improving the results of PI and SS helps to increase the performance of NLP systems, such as statistical machine translation (Madnani et al., 2007) and plagiarism detection (Barrón-Cedeño et al., 2013). PI and SS have been studied intensively for formal

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 3553–3564, Osaka, Japan, December 11-17 2016.

text with important results as shown in (Par, 2016). As social media text is usually very short (e.g., 140-character limit for Twitter) and noisy (flexible nature of personal communication), many NLP systems suffer from the large degree of spelling, syntactic and semantic variants, for example, "ICYMI"= "In case you missed it" or "b/c I love u" = "Because I love you". Traditional approaches have been studied intensively and proved not to work well for social media text (Zanzotto et al., 2011). A few preliminary results have shown that the shortness and noisiness of social media text have significantly decreased the performance of PI (Zanzotto et al., 2011; Xu et al., 2013) and SS tasks (Guo and Diab, 2012; Dang et al., 2015a). In this paper, we proposed a Topic-based Latent Semantic Analysis (TLSA) approach for the SS task, which assigns a semantic similarity score between two social media texts. Next, we use this similarity score to determine if two texts are a paraphrase of each other.

Latent Semantic Analysis (LSA) has been widely used for semantic text similarity tasks because of its simplicity and efficiency (Landauer et al., 1998). LSA has been used as a strong benchmark in the Microsoft Research sentence completion challenge (Zweig and Burges, 2011) and its baseline has outperformed a few state-of-the-art neural network models (Mikolov et al., 2013). However, LSA has its own drawbacks. Its models are trained on a large corpus where words in the same document have a stronger relationship. This does not consider how close two words are in a text ("apple" and "fruit" are closer in the 5-gram "apple is a fruit" instead of a whole document) (Hofmann, 1999). Another example is two topics "Barack Obama" and "Hillary Clinton" have a different meaning in two contexts "2012 US presidential race" and "2016 US presidential race". In the first one, they are opponents, while in the second one, "Barack Obama" endorsed "Hillary Clinton". In addition, LSA is usually trained on a whole corpus. This makes it not scalable with an intrinsic, dynamic, and large-scale nature of social network data. To address this issue, we proposed an approach to train an LSA model that considers the topic being discussed. This proposed LSA model is trained on word 5-grams instead of whole documents. The proposed TLSA method achieved the best result for the SS task and is more scalable compared to other LSA models. Combining TLSA with sentiment analysis, the proposed approach also achieved the best result for PI task in SEMEVAL 2015. These are the contributions of our paper:

1. We create a new word n-gram (1-5) social network corpus from 1.65 billion comments of Reddit[1]. This corpus has two distinctive characteristics that are useful for social media applications: temporal and large-scale social media text.

2. We implement a cloud-based visualization interface so that end users can query and analyze the social media n-grams in real time.

3. We propose TLSA[2], a Topic-based Latent Semantic Analysis model that is trained on word 5-grams from social media text. To the best of our knowledge, there is no similar work that employs a topic-based approach using LSA for PI and SS tasks for social media text.

4. We combine TLSA with sentiment analysis, which outperforms the state-of-the-art unsupervised and semi-supervised methods in SEMEVAL 2015: Paraphrase and Semantic Similarity in Twitter tasks.

2 Related Work

2.1 Corpus-Based algorithms

Corpus-based machine learning algorithms have an advantage over knowledge-based ones as they do not involve in human which can be expensive. The Google web 1T n-gram corpus (Brants and Franz, 2006) included all words appearing on the web in January, 2006 and is available in English and 10 European Languages (Brants and Franz, 2009). This corpus has been used for text relatedness (Islam et al., 2012) and linguistic steganography (Chang and Clark, 2010). The WaCky corpus of more than one billion

[1]Reddit n-gram temporal corpus - https://web.cs.dal.ca/~anh/?page_id=1699

[2]Topic-based Latent Semantic Analysis - http://cgm6.research.cs.dal.ca:8080/RedditFileDownload/tlsa.html

words from three languages, English, German, and Italian was introduced in 2009 by (Baroni et al., 2009). It has been used in bilingual lexicography (Ferraresi et al., 2010) and translators (Pecina et al., 2012). In 2010, Microsoft Web n-gram corpus provided all the word n-grams that are indexed by Bing search engine and provided through an XML web service (Wang et al., 2010b). Some notable usage includes textbox enriching (Agrawal et al., 2010) and social media language study (Liu et al., 2012). Google Book n-gram corpus (Michel et al., 2011), introduced in 2012, includes all word n-grams found in Google book corpus from 1505 to 2008. Due to its yearly temporal characteristics, it has been used to study the changing psychology of culture (Greenfield, 2013), concepts of happiness (Oishi et al., 2013), and mapping book to time (Islam et al., 2015). Twitter n-gram corpus (Herdağdelen and Baroni, 2011) only provides a small subset of social media n-grams in Twitter. As Twitter does not allow researchers to share full text of Tweets as a large corpus, it is not possible to collect, create and share terabyte-scale n-gram corpus for Tweets. Unlike Twitter, Reddit implements an open data policy and users can query any posted data on the website. Although OSNs have been studied intensively in recent years, there is no existing corpus that could be shared and provide insights from massive social network text. To the best of our knowledge, this new corpus is the first large-scale n-gram corpus that provides n-grams with a temporal feature (monthly) that is designed specifically for massive user-generated social media text.

2.2 Paraphrase Identification and Semantic Similarity

A summary of all the existing state-of-the-art paraphrase identification algorithms for traditional texts (e.g., newswire) using the Microsoft Research Paraphrase Corpus (MRPC) is in (Par, 2016). Although supervised approaches, such as typical machine learning classifiers using various feature sets (Das and Smith, 2009; Ji and Eisenstein, 2013) and semantic text similarity (Blacoe and Lapata, 2012; Madnani et al., 2012), achieved the best results, unsupervised methods using explicit semantic space (Hassan and Mihalcea, 2011), vector-based similarity (Milajevs et al., 2014), and WordNet similarity with matrix (Fernando and Stevenson, 2008) also attained comparable results. With the increasing popularity of OSNs, researchers started to focus on the importance of developing paraphrase identification for social media text (Zanzotto et al., 2011; Xu et al., 2013; Guo and Diab, 2012). The results and findings support the hypothesis that informal language in social media with a high degree of lexical variations has posed serious challenges to both tasks. In this paper, our focus is not the general PI or SS tasks but concentrates on the domain of social media.

The SemEval-2015 task 1 is the first competition that focuses on Paraphrase Identification and Semantic Similarity for social media text. There were 19 and 14 teams that participated in the PI and SS tasks, respectively. Most teams used supervised approach, for example, typical machine learning classifiers (Eyecioglu and Keller, 2015), neural networks (Xu et al., 2015), align and penalize architecture, semantic relatedness (van der Goot and van Noord, 2015). Two teams used unsupervised approaches (Orthogonal Matrix Factorization (Guo et al., 2014) and pre-trained word and phrase vectors on Google News dataset (Xu et al., 2015)) and one team uses semi-supervised approach that combines several word measures built from Rovereto Twitter n-gram corpus (Herdağdelen and Baroni, 2011). Our proposed approach will be compared and evaluated against these unsupervised and semi-supervised approaches.

Lately, large corpora are being used for the machine learning tasks. LSA has been widely used for paraphrase identification and semantic text analysis (Hassan and Mihalcea, 2011). (Guo and Diab, 2012) proposed Weighted Textual Matrix Factorization (WTMF), which is a novel latent model that captures the contextual meanings of words in sentences based on internal term-sentence matrix. This model uses both knowledge-based and large-scale corpus-based techniques to learn word representation. Our work uses the new corpus and introduces a novel approach to learn word representation that is dependent on the topic being discussed.

3 The New Reddit Temporal N-gram Corpus

We have created a word n-gram (1-5) corpus of 1.65 billion Reddit comments from October, 2007 until August, 2016 (Baumgartner, July 2015) using high performance distributed processing models on a cluster of 256 nodes with 16TB of shared memory. Most of the comments are in the English language.

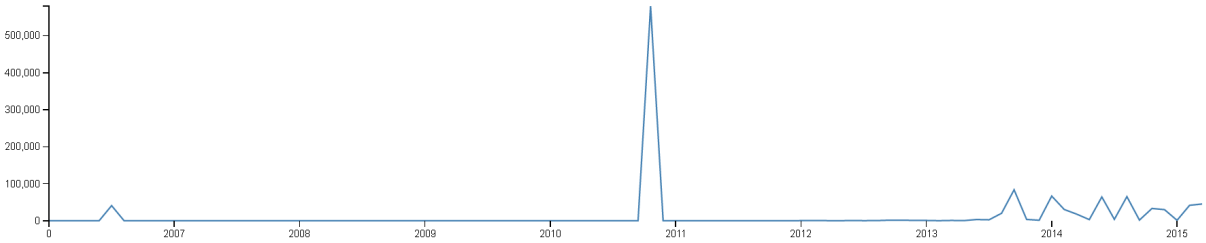

Figure 1: The frequency count of unigram "ISIS" in Reddit from 2007 to 2016. The x-axis represents the year while the y-axis shows the frequency count per month. The highest peak in the graph represents the rise of "ISIS" in October, 2010 following the outbreak of the Syrian Civil War in August, 2010.

Each comment is separated into sentences, and each sentence is tokenized using Lucence (Apache). All the comments are lowercased. The Reddit comments are close to 2TB of text containing 135 billion sentences.

Each entry in the Reddit temporal n-gram corpus is an n-gram (1-5) and its frequency, month, and year from October, 2007 to August, 2016. The size of the corpus is 2.6 TB uncompressed. We show an example of each n-gram (1-5) in Table 1. This corpus can be accessed through downloadable files and a JSON web service which will return the frequency of an n-gram for each month. In addition, we implement a cloud-based visualization interface so that end users can query and analyze the social media n-grams in real time as shown in Figure 1. As our n-gram corpus is time-dependent, we also count the total number of occurrences of each n-gram for comparison with other corpora. A detailed statistical comparison between the new corpus and some other existing corpora is shown in Table 2. Although Microsoft Web n-gram is the largest n-gram corpus, they do not provide all the data to end users. Analyzing the whole Microsoft n-grams corpus is not practical through an XML web service. The Reddit temporal n-gram corpus is much larger than the Google Web 1T n-gram corpus. One of the reasons is that Google Web 1T n-gram corpus only keeps unigrams with more than 200 frequency counts and other n-grams with more than 40 frequency counts. After analyzing the Google Web corpus, we found that although our corpus has a larger vocabulary than the Google one, the frequency count for each n-gram is lower. This confirms the noise and shortness hypothesis of social media text. We decide to keep all the raw n-grams of the new corpus to preserve these characteristics of social media text. We illustrate in Figure 2 how the Reddit temporal n-gram corpus shows the evolution of the word "ISIS".

(a) (b)

Figure 2: An example of word cloud showing the context words of "ISIS" in the 5-grams of the corpus before and after August, 2010. a) "ISIS" is mainly discussed as an Egyptian god before August, 2010, b) "ISIS" means the Islamic State in Iraq and Syria after August, 2010.

Table 1: Examples of n-grams (1-5) of the newly created corpus about the topic "Donald Trump". Each entry includes the word (n-gram), its frequency, its month, and its year from October 2007 to August 2016.

Reddit temporal n-gram corpus	word	frequency	year	month
1-gram	trump	981	2015	01
2-gram	trump apprentice	31	2015	01
3-gram	donald trump battle	16	2015	01
4-gram	donald trump ignorant tweet	8	2015	01
5-gram	take donald trump advice in	2	2015	01

Table 2: Statistical comparison between Reddit temporal n-gram corpus and its counterparts.

Corpus	1-gram	2-gram	3-gram	4-gram	5-gram
Google web 1T n-gram corpus	13.5M	314M	977M	1.3B	1.12B
Microsoft web n-gram corpus	1.2B	11.7B	60.1B	148.5B	237B
Reddit temporal n-gram corpus	170.2M	1.2B	6.7B	18.4B	30.1B

4 Topic-based Latent Semantic Analysis

We first formulate the approach for TLSA. Consider a list of topics $T = \{T_1, T_2, ..., T_l\}$ and each topic T_i has a list of pairs of Tweets $P = \{(t_{11}, t_{12}), (t_{21}, t_{22}), ..., (t_{m1}, t_{m2})\}$ where each pair is evaluated for PI and SS tasks. For each topic T_i, we construct a list of unigrams $O = \{o_1, o_2, ..., o_p\}$ from P and a list of 5-grams $F = \{f_1, f_2, ..., f_q\}$ from Reddit temporal n-gram corpus where f_i contains the topic T_i. Next, we construct the unigram/5-gram matrix X from O and F.

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & \dots & x_{1q} \\ x_{21} & x_{22} & x_{23} & \dots & x_{2q} \\ \dots & \dots & \dots & \dots & \dots \\ x_{p1} & x_{p2} & x_{p3} & \dots & x_{pq} \end{bmatrix}$$

where each row r_i represents the occurrence of a unigram term u_i to all 5-grams in F and x_{ij} describes the occurrence of unigram o_i in a 5-gram f_j plus the frequency of the 5-gram f_j in the Reddit temporal n-gram corpus. This matrix considers both the relation between a word with other words in a 5-gram and with the frequency of this 5-gram in the corpus. Next, we decompose matrix X using a Singular Value Decomposition (SVD):

$$X = U\Sigma V^T$$

$$= \begin{pmatrix} u_1 & \cdots & u_r & u_{r+1} & \cdots & u_p \\ | & & | & | & & | \\ | & & | & | & & | \\ \underbrace{\qquad\qquad}_{\text{col}(A)} & & & \underbrace{\qquad\qquad}_{\text{null}(A)} & & \end{pmatrix} \begin{pmatrix} \sigma_1 & & & & & \\ & \ddots & & & & \\ & & \sigma_r & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0 \end{pmatrix} \begin{pmatrix} \text{---} & v_1^T & \text{---} \\ & \cdots & \\ \text{---} & v_r^T & \text{---} \\ \text{---} & v_{r+1}^T & \text{---} \\ & \cdots & \\ \text{---} & v_q^T & \text{---} \end{pmatrix} \begin{matrix} \\ \\ \} \text{row}(A) \\ \\ \\ \} \text{null}(A) \\ \\ \end{matrix}$$

where Σ is a diagonal matrix that contains the singular values in descending values. U and V are orthogonal matrices that contain the left and right singular vectors respectively.

Next, for each sentence s_i in topic T_i, we construct a vector $\vec{v}$ which represents the occurrence of s_i in the list of unigrams of topic T_i. This vector is translated into a sentence vector representation by the following formula:

$$\vec{v} = \vec{v} * U_k * S_k$$

where k is the chosen k singular values which show the dimensions with the greatest variance between words and documents (the value of k is explained in Section 6.3). Finally, the semantic similarity between two sentences is calculated using the cosine similarity between their vectors.

Due to the enormous size of the Reddit temporal n-gram corpus, selecting the related 5-grams for each topic is not feasible using a traditional relational database system. We tried to load our data into IBM Netezza data warehouse but the query time was not reasonable for a real-time system. We load all the corpus data to Google Bigquery. For each topic T_i, we query all the related 5-grams f_i using Google Bigquery regular expression "word like $(\%T_i\%)$" where $\%$ represents the wild card search. After constructing matrix X, we use Microsoft Azure Apache Spark for SVD decomposition. A summary of the proposed approach is shown in Figure 3.

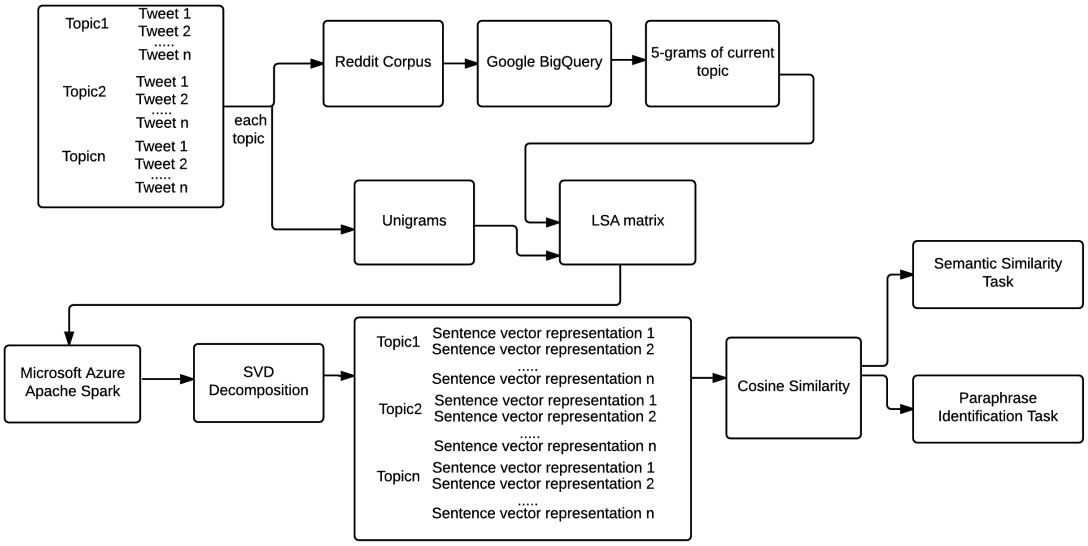

Figure 3: The proposed Topic-based Latent Semantic Analysis using distributed parallel computing, Google BigQuery, and Microsoft Azure Apache Spark. The semantic similarity between two sentences is computed with regard to a specific topic being discussed in two sentences.

5 Evaluations of Paraphrase Identification and Semantic Similarity for Social Media Text

To evaluate the performance of TLSA algorithm, we use the PIT-2015 Twitter dataset (Xu et al., 2014). Although this approach uses PIT-2015 dataset for evaluation, it can be extended to any general topic-based datasets. The PIT-2015 dataset includes 17,790 sentence pairs for training and 972 test sentence pairs which were annotated and developed by (Xu et al., 2014). The dataset was constructed from Twitter data and has intrinsic characteristics from social network data: (i) opinionated and colloquial sentences from realistic social media text; (ii) lexically diverse pairs of sentences for paraphrases; and (iii) sentences that seem lexically similar but semantically dissimilar (Xu et al., 2015). Example pairs of sentences for paraphrase, non-paraphrase, and debatable cases are shown in Table 3. The detailed statistics of this ground-truth dataset is shown in Table 4. Each sentence is processed with tokenization, part-of-speech and named entity tags and each sentence pair is annotated by experts. In the test set, there

Table 3: Examples of Paraphrase Identification and Semantic Similarity sentence pairs. All three sentence pairs are about the movie "8 Mile" which is a topic for TLSA. A sentence pair is a paraphrase if its Pearson Correlation score is above 0.6. A sentence pair is a non-paraphrase if its Pearson Correlation score is below 0.6. A sentence pair is debatable if its Pearson Correlation score is equal to 0.6.

Topic	Paraphase	Sentence 1	Sentence 2
8 mile	True	The Ending to 8 Mile is my fav part of the whole movie	Those last 3 battles in 8 Mile are THE shit
8 mile	False	All the home alones watching 8 mile	The last rap battle in 8 Mile nevr gets old ahah
8 mile	Debatable	8 mile is just a classic	After watching 8 mile I feel like such a thug

are 972 sentence pairs collected from Twitter in 20 trending topics between May 13th and June 10th, 2013. As mentioned in (Das and Smith, 2009), some algorithms may work well specifically for MRPC because of its imbalanced nature (lack of non-paraphrases). PIT-2015 Twitter dataset is more balanced as it contains 70% non-paraphrases and the 34% paraphrases.

Table 4: PIT-2015 Twitter dataset. The test data is more balanced than MRPC as it has a higher percentage of non-paraphrase sentence pairs. The unsupervised TLSA only uses the test data for evaluation.

	Sent Pairs	Paraphase	Non-paraphrase	Debatable
Train	13063	3996 (30.6%)	7534 (57.7%)	1533 (11.7%)
Test	972	175 (18.0%)	663 (68.2%)	134 (13.8%)

5.1 Task 1 - Paraphrase Identification and Evaluation Metrics

For a specific topic, given two sentences, the system has to determine if two sentences have the same or similar meaning and discuss the same topic. For two non-paraphrase sentence pairs, the sentence pair discussing the same topic has a higher score than the sentence pair discussing an unrelated topic. Precision, recall, and F1 (harmonic mean of precision and recall) are used as evaluation metrics.

5.2 Task 2 - Semantic Similarity and Evaluation Metrics

For a specific topic, given two sentences, the system has to give a score between 0 (no relation) and 1 (semantic equivalence) to represent their semantic equivalence. For two sentence pairs, the sentence pair discussing the same topic has a higher semantic similarity score than the sentence pair discussing an unrelated topic. Pearson correlation is used as an evaluation metric.

6 Evaluation

6.1 Baselines

We used first two baselines from (Xu et al., 2015) and introduced two new baselines that are more related to the proposed corpus-based and topic-based LSA.

Random: Each sentence pair is assigned a random real semantic similarity score between [0, 1]. For PI task, this baseline applies 0.5 as a cutoff (paraphrase if semantic similarity score is above 0.5).

Weighted Matrix Factorization (WTMF): This baseline uses the state-of-the-art unsupervised method of (Guo and Diab, 2012). It not only considers the semantic space of words presenting in the data but also missing words from the sentences. This feature is designed specifically for short texts in social media. Finally, the value 0.5 is used as a cutoff for the PI task.

Random 5-gram: This baseline determines whether introducing the use of topics in LSA improves the accuracy of both PI and SS tasks for SEMEVAL 2015. To construct matrix X, we select random

5-grams from the Reddit temporal n-gram corpus with the same size of the 5-grams that contain the topic.

Google Tri-gram Method (GTM): Google Tri-gram Method (Islam et al., 2012) assigns a semantic similarity score between two sentences using the unigrams and trigrams of the Google Web 1T corpus. We also use 0.5 as a cutoff for the PI task.

6.2 SEMEVAL 2015 Unsupervised and Semi-supervised Methods

Columbia: This method used Orthogonal Matrix Factorization to compute a representation vector for each sentence (Guo et al., 2014) and then computes a similarity score based on these vectors (Xu et al., 2015).

Yamraj: This method learned sentence vectors from Google News dataset (about 100 billion words) and Wikipedia articles. Cosine distance is used to compute the vector similarity scores.

MathLingBp: This method exploits the use of the align-and-penalize architecture of (Han et al., 2013) and adopts the use of several word similarity metrics using a semi-supervised approach (Xu et al., 2015).

6.3 Experimental Results

First, we compare the performance of TLSA with various parameters, such as the number of singular values and the dimensionality of the 5-grams. For SS task, we achieved the best result for SS task when the singular value k is equal to 80 with an increasing 5-gram dimensionality size as shown in Figure 4. In addition, for SS task, the Pearson correlation is not improving when the number of 5-grams is above 1M.

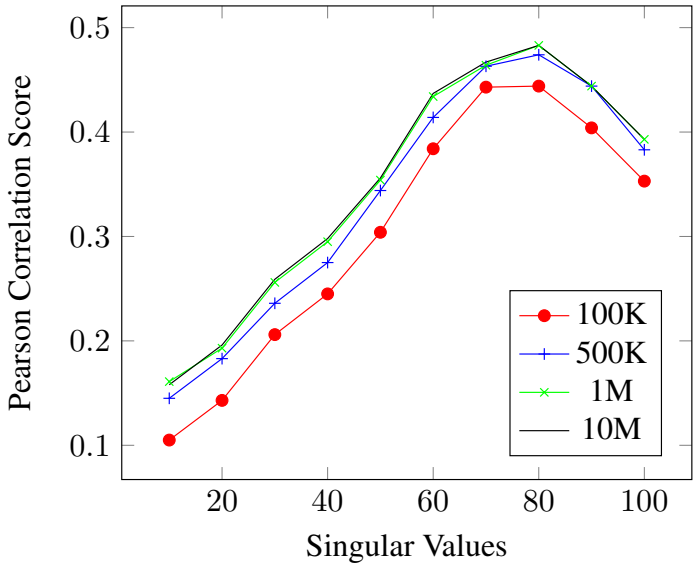

Figure 4: For SS task, Pearson correlation score with an increasing singular values and 5-gram dimensionality. TLSA achieves the best Pearson correlation score for k = 80 and the dimensionality of 5-grams = 1M.

6.3.1 Topic-based LSA versus Baselines and other Methods

This section compares the proposed approach with the baselines and SEMEVAL 2015 unsupervised and semi-supervised methods. As shown in Table 5, TLSA achieved the best result for the SS task (Pearson correlation) compared with all the baselines and compared methods. This means that training an LSA model using topic-based 5-gram helps increase the result of PI and SS tasks. For the PI task, observing that the semantic similarity scores for sentence pairs are either very high or very low, we tried two cutoffs 0.25 and 0.5 (SEMEVAL 2015 allows two runs per team) and TLSA outperforms all the baselines. With a low cutoff value, TLSA achieves a high precision and a low recall. To improve the PI results, we assumed that two sentences are paraphrases only if they have the same sentiment scores

(e.g., both are positives or negatives). Based on this assumption, each sentence is assigned a sentiment score using OpenNLP. Adding sentiment analysis to TLSA (i.e., TLSA & Sentiment) outperforms all the baselines and compared methods. Another important observation is that although our unsupervised approach achieves the best results against the baselines and compared methods, its results are still not comparable with human upperbound. This means that improving the results of PI and SS tasks for social media text using an unsupervised approach is still a challenge for researchers.

Table 5: TLSA results with other baselines and compared methods. Combining TLSA with sentiment analysis achieves the best result for both PI and SS tasks.

Methods / *Baselines*	Paraphrase Identification			Semantic Similarity			
	F1	Precision	Recall	Pearson	maxF1	maxPrec	maxRecall
Human Upperbound	0.823	0.752	0.0908	0.735	–	–	–
TLSA & Sentiment	**0.591**	0.764	0.480	**0.483**	0.582	0.761	0.472
COLUMBIA	0.588	0.593	0.583	0.425	0.599	0.623	0.577
TLSA	**0.585**	0.761	0.474	**0.483**	0.585	0.761	0.474
YAMRAJ	0.496	0.725	0.377	0.360	0.542	0.502	0.589
WTMF	0.536	0.450	0.663	0.350	0.587	0.570	0.606
Random 5-gram	0.504	0.716	0.389	0.466	0.564	0.824	0.429
GTM	0.495	0.391	0.674	0.371	0.582	0.761	0.472
Random	0.266	0.192	0.434	0.017	0.350	0.215	0.949

7 Conclusions

In this paper, we introduced Reddit temporal n-gram corpus, which is designed specifically for social media text. We create the corpus using distributed parallel computing and implement a cloud-based visualization interface so that end users can query any n-grams from the corpus. Both the corpus and the interface are publicly available in this URL - Reddit n-gram temporal corpus. This large-scale terabyte corpus includes all the word unigram to 5-gram, and their frequency per month from October, 2007 to August, 2016.

To show the usefulness of this corpus, we propose a novel Topic-based Latent Semantic Analysis approach which exploits the 5-grams of the corpus. The proposed TLSA outperforms all the state-of-the-art unsupervised and semi-supervised methods in SEMEVAL 2015 Task 1 - Semantic Similarity for the PIT-2015 dataset. Combining with sentiment analysis, the proposed approach also achieves the best result for the Paraphrase Identification of SEMEVAL 2015 Task 1. In addition, TLSA is language-independent and scalable for the large-scale nature of social media text.

For future work, we aim to use this corpus to study the linguistic patterns of social media text, for example, finding the meaning of new words in social media. In addition, we plan to integrate this proposed semantic similarity score into our existing work to improve the results of meme clustering tasks (Dang et al., 2015b) and rumour detection and visualization framework (Dang et al., 2016).

Acknowledgment

The research was funded in part by CNPq 487186/2013-3, FAPESP 2014/09599-5 (Brazil), Natural Sciences and Engineering Research Council of Canada, and International Development Research Centre, Ottawa, Canada.

References

Rakesh Agrawal, Sreenivas Gollapudi, Krishnaram Kenthapadi, Nitish Srivastava, and Raja Velu. 2010. Enriching textbooks through data mining. In *Proc. of the First ACM Symposium on Computing for Development*, page 19. ACM.

Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The wacky wide web: a collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226.

Alberto Barrón-Cedeño, Marta Vila, M Antònia Martí, and Paolo Rosso. 2013. Plagiarism meets paraphrasing: Insights for the next generation in automatic plagiarism detection. *Computational Linguistics*, 39(4):917–947.

Jason Baumgartner. July, 2015. Complete public reddit comments corpus. Available: https://archive.org/details/2015_reddit_comments_corpus [Accessed: April 13, 2016].

William Blacoe and Mirella Lapata. 2012. A comparison of vector-based representations for semantic composition. In *Proc. of the 2012 Joint Conference on EMNLP and Computational Natural Language Learning*, pages 546–556. ACL.

Thorsten Brants and Alex Franz. 2006. {Web 1T 5-gram Version 1}.

Thorsten Brants and Alex Franz. 2009. Web 1t 5-gram, 10 european languages version 1. *Linguistic Data Consortium, Philadelphia*.

Ching-Yun Chang and Stephen Clark. 2010. Linguistic steganography using automatically generated paraphrases. In *HLT- NAACL*, pages 591–599.

Charles LA Clarke, Gordon V Cormack, M Laszlo, Thomas R Lynam, and Egidio L Terra. 2002. The impact of corpus size on question answering performance. In *ACM SIGIR*, pages 369–370. ACM.

Anh Dang, Raheleh Makki, Abidalrahman Moh'd, Aminul Islam, Vlado Keselj, and Evangelos E Milios. 2015a. Real time filtering of tweets using wikipedia concepts and google tri-gram semantic relatedness. In *TREC*.

Anh Dang, Abidalrahman Moh'd, Anatoliy Gruzd, Evangelos Milios, and Rosane Minghim. 2015b. A visual framework for clustering memes in social media. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, ASONAM '15, pages 713–720, New York, NY, USA. ACM.

Anh Dang, Abidalrahman Moh'd, Evangelos Milios, and Rosane Minghim. 2016. What is in a rumour: Combined visual analysis of rumour flow and user activity. In *Proceedings of the 33rd Computer Graphics International*, pages 17–20. ACM.

Dipanjan Das and Noah A Smith. 2009. Paraphrase identification as probabilistic quasi-synchronous recognition. In *Proc. of the Joint Conference of the 47th Annual Meeting of the ACL and AFNLP: Volume 1-Volume 1*, pages 468–476. ACL.

Asli Eyecioglu and Bill Keller. 2015. Asobek: Twitter paraphrase identification with simple overlap features and svms. *SemEval*.

Samuel Fernando and Mark Stevenson. 2008. A semantic similarity approach to paraphrase detection. In *Proc. of the 11th Annual Research Colloquium of the UK Special Interest Group for Computational Linguistics*, pages 45–52. Citeseer.

Adriano Ferraresi, Silvia Bernardini, Giovanni Picci, and Marco Baroni. 2010. Web corpora for bilingual lexicography: a pilot study of english/french collocation extraction and translation. *Using Corpora in Contrastive and Translation Studies. Newcastle: Cambridge Scholars Publishing*, pages 337–362.

Patricia M Greenfield. 2013. The changing psychology of culture from 1800 through 2000. *Psychological Science*, 24(9):1722–1731.

Weiwei Guo and Mona Diab. 2012. Modeling sentences in the latent space. In *Proc. of the 50th Annual Meeting of the ACL: Long Papers-Volume 1*, pages 864–872. ACL.

Weiwei Guo, Wei Liu, and Mona T Diab. 2014. Fast tweet retrieval with compact binary codes. In *COLING*, pages 486–496. Citeseer.

Lushan Han, Abhay Kashyap, Tim Finin, James Mayfield, and Jonathan Weese. 2013. Umbc ebiquity-core: Semantic textual similarity systems. In *Proc. of the Second Joint Conference on Lexical and Computational Semantics*, volume 1, pages 44–52.

Samer Hassan and Rada Mihalcea. 2011. Semantic relatedness using salient semantic analysis. In *AAAI*.

Amaç Herdağdelen and Marco Baroni. 2011. Stereotypical gender actions can be extracted from web text. *Journal of the American Society for Information Science and Technology*, 62(9):1741–1749.

Thomas Hofmann. 1999. Probabilistic latent semantic analysis. In *Proc. of the Fifteenth conference on Uncertainty in Artificial Intelligence*, pages 289–296. Morgan Kaufmann Publishers Inc.

Aminul Islam, Evangelos Milios, and Vlado Kešelj. 2012. Text similarity using google tri-grams. In *Advances in Artificial Intelligence*, pages 312–317. Springer.

Aminul Islam, Jie Mei, Evangelos E Milios, and Vlado Kešelj. 2015. When was macbeth written? mapping book to time. In *Computational Linguistics and Intelligent Text Processing*, pages 73–84. Springer.

Yangfeng Ji and Jacob Eisenstein. 2013. Discriminative improvements to distributional sentence similarity. In *EMNLP*, pages 891–896.

Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. 2010. What is twitter, a social network or a news media? In *WWW*, pages 591–600. ACM.

Thomas K Landauer, Peter W Foltz, and Darrell Laham. 1998. An introduction to latent semantic analysis. *Discourse Processes*, 25(2-3):259–284.

Chen Li and Yang Liu. 2015. Improving named entity recognition in tweets via detecting non-standard words. In *ACL*.

Fei Liu, Fuliang Weng, and Xiao Jiang. 2012. A broad-coverage normalization system for social media language. In *ACL*, pages 1035–1044.

Nitin Madnani, Necip Fazil Ayan, Philip Resnik, and Bonnie J Dorr. 2007. Using paraphrases for parameter tuning in statistical machine translation. In *Proc. of the Second Workshop on Statistical Machine Translation*, pages 120–127. ACL.

Nitin Madnani, Joel Tetreault, and Martin Chodorow. 2012. Re-examining machine translation metrics for paraphrase identification. In *Proc. of the 2012 Conference of the NAACL: Human Language Technologies*, pages 182–190. ACL.

Jean-Baptiste Michel, Yuan Kui Shen, et al. 2011. Quantitative analysis of culture using millions of digitized books. *Science*, 331(6014):176–182.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *ArXiv Preprint ArXiv:1301.3781*.

Dmitrijs Milajevs, Dimitri Kartsaklis, Mehrnoosh Sadrzadeh, and Matthew Purver. 2014. Evaluating neural word representations in tensor-based compositional settings. *ArXiv Preprint ArXiv:1408.6179*.

Peter Norvig. 2008. Statistical learning as the ultimate agile development tool. In *CIKM*.

Shigehiro Oishi, Jesse Graham, Selin Kesebir, and Iolanda Costa Galinha. 2013. Concepts of happiness across time and cultures. *Personality and Social Psychology Bulletin*, 39(5):559–577.

Par. 2016. Paraphrase identification (state of the art) @ONLINE. Available: http://aclweb.org/aclwiki/index.php?title=Paraphrase_Identification_(State_of_the_art) [Accessed: July 15, 2016].

Pavel Pecina, Antonio Toral, et al. 2012. Domain adaptation of statistical machine translation using web-crawled resources: a case study. In *EAMT*, pages 145–152.

Rob van der Goot and Gertjan van Noord. 2015. Rob: Using semantic meaning to recognize paraphrases. *SemEval*.

Kuansan Wang, Xiaolong Li, and Jianfeng Gao. 2010a. Multi-style language model for web scale information retrieval. In *ACM SIGIR*, pages 467–474. ACM.

Kuansan Wang, Christopher Thrasher, Evelyne Viegas, Xiaolong Li, and Bo-june Paul Hsu. 2010b. An overview of microsoft web n-gram corpus and applications. In *NAACL HLT 2010 Demonstration Session*, pages 45–48.

Wei Xu, Alan Ritter, and Ralph Grishman. 2013. Gathering and generating paraphrases from twitter with application to normalization. In *Proc. of the Sixth Workshop on Building and Using Comparable Corpora*, pages 121–128. Citeseer.

Wei Xu, Alan Ritter, Chris Callison-Burch, William B Dolan, and Yangfeng Ji. 2014. Extracting lexically divergent paraphrases from twitter. *Transactions of the ACL*, 2:435–448.

Wei Xu, Chris Callison-Burch, and William B Dolan. 2015. Semeval-2015 task 1: Paraphrase and semantic similarity in twitter (pit). *SemEval*.

Fabio Massimo Zanzotto, Marco Pennacchiotti, and Kostas Tsioutsiouliklis. 2011. Linguistic redundancy in twitter. In *EMNLP*, pages 659–669. ACL.

Yating Zhang, Adam Jatowt, Sourav S Bhowmick, and Katsumi Tanaka. 2015. Omnia mutantur, nihil interit: Connecting past with present by find-ing corresponding terms across time. In *ACL*, pages 645–655.

Geoffrey Zweig and Christopher JC Burges. 2011. The microsoft research sentence completion challenge. Technical report, Technical Report MSR-TR-2011-129, Microsoft.