Context Tailoring for Text Normalization

Seniz Demir
TUBITAK BILGEM
Kocaeli, TURKEY
seniz.demir@tubitak.gov.tr

Abstract

Language processing tools suffer from significant performance drops in social media domain due to its continuously evolving language. Transforming non-standard words into their standard forms has been studied as a step towards proper processing of ill-formed texts. This work describes a normalization system that considers contextual and lexical similarities between standard and non-standard words for removing noise in texts. A bipartite graph that represents contexts shared by words in a large unlabeled text corpus is utilized for exploring normalization candidates via random walks. Input context of a non-standard word in a given sentence is tailored in cases where a direct match to shared contexts is not possible. The performance of the system was evaluated on Turkish social media texts.

1 Introduction

Social media has been an integral part of personal communication in the last decade. Everyday, people willingly produce millions of multilingual texts since they are free in the way how they express themselves. Grammatical rules and language structures do not have to be all in place, even the words might not be properly written. However, free-style writing and ever-growing nature of the language hinder the utility of social media texts in computational linguistics studies. For instance, non-standard words (e.g.,"comin soon" for "coming soon") and phonetic substitutions (e.g., "4u" for "for you") that are often seen in social media texts degrade the performance of many NLP tools such as parsers and named entity recognizers (Foster et al., 2011; Kucuk and Steinberger, 2014). Being trained on formal texts is the major drawback of these tools in processing ill-formed texts. In addition, different social media genres have their own characteristics and various factors (e.g., demographic background and age) trigger linguistic changes in social media language over time (Eisenstein et al., 2014; Herdagdelen, 2013; Schwartz et al., 2013).

Normalizing ill-formed texts is a promising preprocessing step to address experienced accuracy drops in existing NLP tools. This paper proposes a normalization system that transforms non-standard out of vocabulary (OOV) words into their standard in vocabulary (IV) forms. We argue that contextual and lexical characteristics of words are of the greatest importance in normalization. To encode contextual similarities of words, a bipartite graph acquired from a corpus of formal and informal texts is utilized. The first bipartite of the graph consists of words that appear in the contexts represented by the second bipartite. This graphical representation not only captures the way how words are used in real texts but also presents lexical variants of the same word in similar contexts.

A non-standard word might be restored to different words depending on its context in a sentence. In this work, one consideration that is exploited in determining the right form of a non-standard word is its input context and the similarity of this context to other contexts captured by the underlying graph. At the core of our normalization system lies a novel use of a tailoring approach on input contexts. Tailored input contexts are shown to be effective in finding

Proceedings of the 2016 Workshop on Graph-based Methods for Natural Language Processing, NAACL-HLT 2016, pages 6–14,
San Diego, California, June 17, 2016. ©2016 Association for Computational Linguistics

similar contexts in the graph. To normalize a non-standard word, our system determines contextually similar candidate normalizations by performing random walks through similar contexts. Additional normalization candidates are extracted from a word lexicon by measuring their lexical distances to the ill-formed word via a well-known similarity metric. We evaluated our normalization system on 715 Turkish social media sentences with 1190 ill-formed words in total and our promising results revealed that we improve on state-of-the-art in Turkish.

2 Related Work

Text normalization has been extensively studied in the literature. Noisy channel model where posterior probabilities are used to find the most probable form of a non-standard word has pioneered normalization research (Brill and Moore, 2000; Tautanova and Moore, 2002; Choudhury et al., 2007; Cook and Stevenson, 2009). Handling the transformation in normalization as a translation from an ill-formed text to a formal text has also been experimented (Aw et al., 2006; Kaufmann and Kalita, 2010). These earlier studies have relied on hand annotated data and proper categorization of non-standard words.

A recent work has used a letter transformation approach where the generation of a non-standard word from a standard word is modeled at the character level. Visual priming and string/phonetic similarity have also been incorporated into the transformation model (Liu et al., 2011; Liu et al., 2012). Generating a confusion set of in-vocabulary word candidates for an out-of-vocabulary word and selecting the best candidate according to lexical string similarity and contextual features have been studied (Han et al., 2013). More recently, distributed word representations (Kumar and Sridhar, 2015) and recurrent neural networks (Chrupala, 2014) have been used to address the normalization problem.

Our work is closely related to the latter line of research which has used graphical representations to capture contextual similarities between words. The work of Sönmez and Özgür (2014) has addressed the normalization problem via a word-association graph where the graph represents POS tags of words and their relative positions to each other in written texts. The proposed approach has achieved a 94.1% precision on a shared English dataset by taking contextual, grammatical, and lexical features of words into account. Our work differs from their approach in several aspects. First, we take into account all words in the input context of an ill-formed word at once (if possible) rather than aggregating individual relations of an ill-formed word with each word in its context. Second, we do not benefit from the part-of-speech of an ill-formed word during normalization but it plays an important role in their unsupervised approach. In another approach, the underlying graph has been used to assess how contextually similar two words are (Hassan and Menezes, 2013). Randomly traversing the graph, a normalization lexicon was induced and later used to generate word confusion sets of non-standard words. This work, which generates the same confusion set for an OOV word in different input contexts, has achieved a precision of 92.43% on English social media texts.

Unfortunately, a few recent studies have focused on text normalization in Turkish. In the cascaded approach (Torunoğlu and Eryiğit, 2014), seven different transformations (e.g., vowel restoration and accent normalization) have been developed to restore OOV words to their IV word forms. The approach has achieved an accuracy of 71% on a set of 600 Turkish tweets. The most recent model (Yıldırım and Yıldız, 2015) has utilized a variety of techniques (e.g., lexical similarity and language model based contextual similarity) to handle different normalization problems and a precision of 80% has been reported on Turkish tweets.

3 Normalization Equivalences

We argue that an ill-formed word and its standard rendering (normalization equivalence) have some characteristics that should be considered in normalization. First, a standard word replaces its ill-formed version in any sentence without a loss in meaning. Therefore, shared contexts where standard and non-standard words both appear give important clues for normalization. Second, transformation from an ill-formed word to different standard words is possible. For instance, the word 'aaba' in the sentence "aaba annem de okulumu sevecek mi?"{I wonder if my mother will also like my school?} should be normalized as 'acaba'{I wonder}. However, that word should be normalized as 'araba'{car} in the sentence "aaba sürmek benim için büyük keyif"{Driving car is a great pleasure for me.}. Thus, input context of an ill-formed word is of great importance and cannot be ignored in normalization. Third, character-based edit operations (i.e, insertion, deletion, or substitution) produce non-standard forms of a word. These operations might be phonetic, graphemic, typographic, etc. (Liu et al., 2012). For instance, the ill-formed word 'ailelerne' can be produced from standard words 'ailelere' (with one insertion) and 'ailelerine' (with one deletion). Therefore, ill-formed words and their standard forms are lexically/phonetically similar.

3.1 Shared Contexts

We treat the context of a word as an n-gram word sequence that has the word at its center. For instance, consider the 5-gram word sequence $w_1w_2w_3w_4w_5$. Here, the context of the word w_3 consists of the two words on its left and the two words on its right $w_1w_2w_4w_5$. Our system uses a bipartite graph to represent words and their contexts. The first bipartite contains words **(word nodes)** and the second bipartite represents contexts where these words appear **(context nodes)**. A word node is connected to a context node with an undirected edge if the word appears in that context. For instance, Figure 1 shows a bipartite graph[1] constructed from a set of 5-gram sequences where an edge weight represents co-occurrence count of the corresponding word and context in the set. A context node might be connected to more than one word node. We refer to these context nodes as "shared contexts" such as the nodes Cntx$_1$ and Cntx$_2$ in Figure 1.

3.2 Context of an Ill-Formed Word

The input context of an ill-formed word is also represented as an n-gram word sequence. In our system, two kinds of contexts are used for an ill-formed word. The central context is an n-gram sequence which has the ill-formed word at its center. However, sliding contexts are those n-gram sequences which have the ill-formed word at any position but the center. An ill-formed word might have an incomplete central context with one or more missing words ($\emptyset$[2]) due to its position in a sentence. One or more sliding contexts are always possible for an ill-formed word with/without missing words. For instance, consider the sentence "bugünn oğlumu gösterisi için hazırlıyıp fotoğraflarını çekeceğim"{Today, I will prepare my son for his show and take his photos}. The contexts extracted for the ill-formed words of the sentence using 5-gram sequences are shown below:

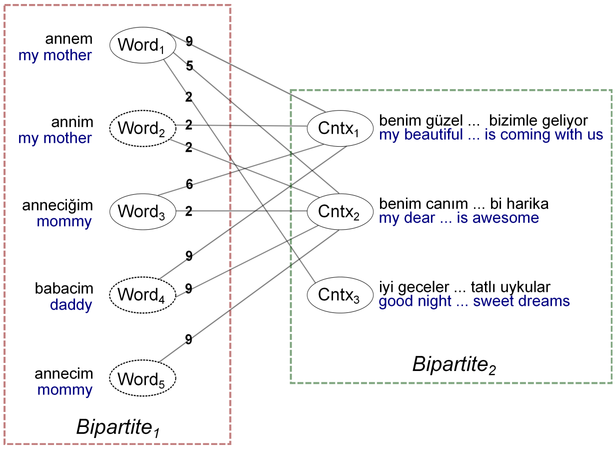

Figure 1: A bipartite graph built from a set of word sequences.

Word: **'bugünn'{today}**
Central context: [$\emptyset$, $\emptyset$, oğlumu, gösterisi]
Sliding contexts: [bugünn, oğlumu, için, hazırlıyıp]

Word: **'hazırlıyıp'{prepare}**
Central context: [gösterisi, için, fotoğraflarını, çekeceğim]
Sliding contexts: [bugünn, oğlumu, için hazırlıyıp], [oğlumu, gösterisi, hazırlıyıp, fotoğraflarını]

3.3 Edit Operations

In our system, only one-to-one word normalizations where an ill-formed word is restored to a single standard word are considered. In such cases, a standard word represents a number of edit operations (e.g., insertion or substitution) once aligned to its ill-formed lexical variant at the character level. We capture the operations that a standard word experience using a well-known lexical similarity metric.

[1] The word nodes Word$_2$, Word$_4$, and Word$_5$ represent OOV words whereas nodes Word$_1$ and Word$_3$ represent IV words.

[2] The symbol $\emptyset$ refers to any word.

4 Normalization Approach

Once a sentence is given, our system analyzes all words using a morphological parser and those that cannot be parsed are accepted as OOV words. For each OOV word, normalization candidates which reflect all characteristics that we consider in this work are explored in the underlying graph. This exploration, performed by a number of random walks, can benefit from three pieces of information: the ill-formed word, its input contexts, and all kinds of information captured by the underlying graph. A word lexicon that consists of standard IV words is also examined to identify candidate normalizations that are lexically similar to the OOV word. Finally, one candidate is selected from among all candidates as the best normalization of the OOV word.

4.1 Contextual Similarity via Random Walks

A random walk starts from a node of a bipartite and consists of a number of sequential steps taken from one bipartite to another (from word node to context node and vice versa). A step cannot be taken within a bipartite since there are no connections between its nodes. Transition probabilities (TP) from the current node are used to determine which node(s) will be visited next.

$$\text{TP}_{ab} = \text{TFreq}^{1,2}_{ab} \Big/ \sum_{\substack{x \in \\ Neighs(a)}} \text{TFreq}^{1,2}_{ax}$$

$$\text{TFreq}^{1}_{ab} = \text{Weight}_{ab} \Big/ \text{Freq(b)}$$

$$\text{TFreq}^{2}_{ab} = \text{Weight}_{ab}$$

TFreq^{1}_{ab} measures the transition frequency from a context node (a) to a word node (b) and TFreq^{2}_{ab} measures the transition frequency from a word node (a) to a context node (b). Weight_{ab} corresponds to the weight of the edge connecting nodes a and b in the graph. Although moving to a common word is penalized by taking into account its frequency in the corpus (Freq(b)), no penalty is applied to contexts shared by many words.

In our system, the maximum number of steps that can be taken in a random walk is limited. A random walk ends when a word node representing a standard word is reached or the maximum number of steps is exhausted without reaching a standard word node. For instance, assume that the graph shown in Figure 1 is a subgraph of a larger bipartite graph where the word nodes have connections to other context nodes that are not shown in the graph. Assume also that two possible random walks of at most 4 steps from Word$_2$ are performed on the graph, which are Word$_2$-Cntx$_1$-Word$_1$ (with two steps) and Word$_2$-Cntx$_1$-Word$_4$-Cntx$_2$-Word$_5$ (with four steps). In this scenario, the first walk ends at Word$_1$ since it represents a standard word but the second walk ends at Word$_5$ since the maximum number of steps is taken without reaching a standard word node.

The node from where a random walk starts **(initial node)** is of great importance in a graph with so many nodes and connections due to the limited number of steps. Since these walks are for identifying candidate normalizations for an ill-formed word, all available information about that word should be utilized in selecting the initial node. It is straightforward to select the node representing ill-formed word as the initial node and to explore normalization candidates by randomly walking from this node. However, this approach would fail in utilizing other valuable information sources, namely input contexts of the word. We argue that input contexts of the ill-formed word should be used in determining the initial node and restricting the space of neighbor nodes to be visited from the initial node. Our approach uses similarities between the central context of the ill-formed word and contexts represented by the graph to determine the initial node.

Two contexts are treated as similar if they are exactly the same or a match can be achieved if any of them is tailored by compromising from its contextual content. A context can be tailored by relaxing constraints put by contained words. In determining the initial node, only central contexts are tailored by replacing one or more contained words with $\emptyset$ symbol, which in turn can match to any word. For instance, the context [yemek, yaparken, fazla, kullanmak]{using ... a lot while cooking} turns to be similar to the context [yemek, yaparken, az, kullanmak]{using ... a little while cooking} if it is tailored to [yemek, yaparken, $\emptyset$, kullanmak]. Relaxing restrictions enables more contexts to be explored in the underlying graph but makes central context less beneficiary for reducing randomness. For instance, consider the central context [Word$_1$, Word$_2$,

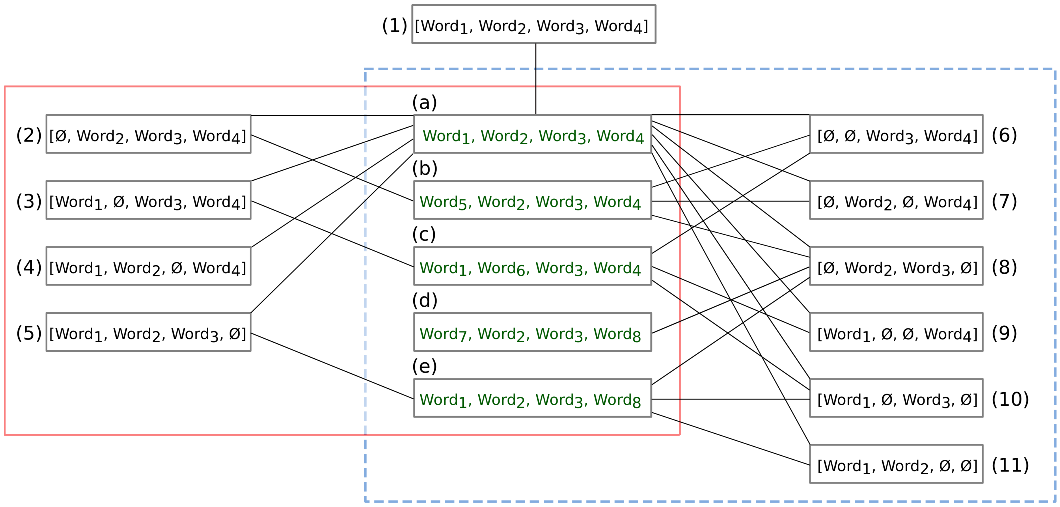
(1) [Word1, Word2, Word3, Word4]
(a)
Word1, Word2, Word3, Word4
(b)
Word5, Word2, Word3, Word4
(c)
Word1, Word6, Word3, Word4
(d)
Word7, Word2, Word3, Word8
(e)
Word1, Word2, Word3, Word8
(2) [Ø, Word2, Word3, Word4]
(3) [Word1, Ø, Word3, Word4]
(4) [Word1, Word2, Ø, Word4]
(5) [Word1, Word2, Word3, Ø]
[Ø, Ø, Word3, Word4] (6)
[Ø, Word2, Ø, Word4] (7)
[Ø, Word2, Word3, Ø] (8)
[Word1, Ø, Ø, Word4] (9)
[Word1, Ø, Word3, Ø] (10)
[Word1, Word2, Ø, Ø] (11)

Figure 2: Context tailoring.

Word$_3$, Word$_4$] shown in Figure 2. This context directly matches Context (a) but not the Contexts (b) - (e). The central context can be tailored to Contexts (2), (3), (4), and (5) by relaxing restrictions put by a single word whereas to Contexts (6), (7), (8), (9), (10), and (11) with two words. The more restrictions are relaxed, the more similar contexts are found (e.g., Context (d)) in the underlying graph.

Our approach first identifies the node representing the ill-formed word and its neighbor context nodes. If any of these connected context nodes matches the central context of the word, it is identified as the initial node. If a direct match is not possible, the central context is tailored in turns until similar contexts are found among the neighbors. At each turn, all possible contexts obtained by relaxing the restriction put by a single word in tailored contexts of the previous turn are used. The neighbor nodes that represent contexts similar to a tailored form of the central context are identified as initial nodes. If a similar context is not found until the tailoring threshold is reached[3], the node that represents the ill-formed word is used as the initial node.

On the other hand, if the ill-formed word does not appear in the underlying graph, its central context is compared to all contexts represented in the graph using the same tailoring methodology. Context nodes that represent similar contexts to tailored central context are identified as initial nodes. If more than one initial node is identified, random walks could be performed either from one or some, or even all nodes depending on the traversal strategy. For instance, assume that an ill-formed word is connected to the Contexts (a) - (e) shown in Figure 2. Initial nodes determined for that word in some representative central contexts are shown below:

- Context: [Word$_1$, Word$_2$, Word$_7$, Word$_4$]
  Initial Node: Context (a)
- Context: [Word$_1$, Word$_2$, Word$_4$, Word$_9$]
  Initial Nodes: Context (a) & Context (e)

For an ill-formed word, a fixed number of random walks are performed starting from the same initial node(s) in order to find its normalization candidates. We treat a standard word where any random walk ends as contextually similar to the ill-formed word. For instance, after the walk Word$_2$-Cntx$_1$-Word$_1$ in Figure 1, the words ‘annim’ and ‘annem’ are considered as contextually similar.

4.2 Positional Similarity

An ill-formed word might not be represented as a word node in the underlying graph if it does not appear at the center of an n-gram word sequence. Nonetheless, its position in a context still provides insights into how that word could be normalized. In a nutshell, we search for other words that are rather used at the position of the ill-formed word in similar contexts. Here, sliding contexts of the ill-formed word are used. For each sliding context,

[3] We limit the number of $\emptyset$ symbols in a context.

the underlying graph is explored using the tailoring methodology in order to find similar contexts and to identify words that appear at the position of the ill-formed word in these contexts. We treat these words as positionally similar to the ill-formed word. For instance, assume that positionally similar words are searched for the word ‘hazırlıyıp’ in the sentence “bugünn oğlumu gösterisi için hazırlıyıp fotoğraflarını çekeceğim”. Assume also that only the contexts shown below are identified as similar to the sliding contexts of that word (given in Section 3.2). In this scenario, positionally similar words to the word ‘hazırlıyıp’ are shown below in bold:

> [bugünn, oğlumu, için, **hazırlayıp**]
> [oğlumu, gösterisi, **süsleyip**, resimlerini]
> [oğlumu, gösterisi, **bezeyip**, fotoğraflarını]

4.3 Lexical Similarity

Lexical similarity of normalization equivalences is as important as their contextual similarity. For an ill-formed word, we also explore a lexicon of standard words as an external resource in order to determine lexically similar words among them. In this work, two well-known metrics, namely Longest Common Subsequence Ratio ($LCSR_{ab}$) and Edit Distance (ED_{ab}) are used to assess how lexically similar two words are (LexSim_{ab}):

$$\text{LexSim}_{ab} = \text{LCSR}_{ab} \;/\; \text{ED}_{ab}$$

$$\text{LCSR}_{ab} = \text{LCS}_{ab} \;/\; \text{Max}\langle \text{Length(a), Length(b)} \rangle$$

For instance, consider the ill-formed word ‘köşlerine’ which can be produced from a standard word ‘köylerine’{to their town} by substituting the ‘y’ character with ‘ş’ or from the word “köşelerine”{to its corners} by dropping the ‘e’ character. Although the ill-formed word has the same edit distance to both words, its lexical similarity is 0.89 to ‘köylerine’ and 0.9 to ‘köşelerine’.

4.4 Methodology

Our normalization methodology separately generates normalization candidates that are (i) contextually similar, (ii) positionally similar, and (iii) lexically similar to an ill-formed word. Contextually and positionally similar candidates whose edit distance to the ill-formed word is greater than a predetermined threshold are discarded. For each case, the remaining candidates are assigned a similarity score between 0 and 1. Contextual similarity score (CS_{ab}) is computed using the hitting time between two nodes (HT_{ab}). In this work, hitting time refers to the minimum number of steps taken to visit one node starting from the other node in any performed random walk[4]:

$$\text{CS}_{ab} = \text{Min}\langle \text{HT}_{ax} \rangle \;/\; \text{HT}_{ab}$$

Frequencies of contexts in the corpus on which the graph is built are used for computing positional similarity scores (PS_{ab}):

$$\text{PS}_{ab} = \text{Freq(Context}_b) \;/\; \text{Max}\langle \text{Freq(Context}_x) \rangle$$

Lexical similarity score (LS_{ab}) is computed using the LexSim formula:

$$\text{LS}_{ab} = \text{LexSim}_{ab} \;/\; \text{Max}\langle \text{LexSim}_{ax} \rangle$$

Finally, normalization scores of all candidates (NS_{ab}) are computed and the candidate with the highest score is selected as the normalized form of the ill-formed word[5]:

$$\text{NS}_{ab} = \lambda_1 \times \text{CS}_{ab} + \lambda_2 \times \text{PS}_{ab} + \lambda_3 \times \text{LS}_{ab}$$

5 Evaluation

In order to evaluate the performance of our language-independent system, we experimented with Turkish social media texts. For the study, we first collected a large corpus of noisy and clean Turkish texts from different resources. The corpus consisted of tweets (∼11GB) that were retrieved via Twitter Streaming API from April to October 2015, 20 million publicly available tweets[6], and clean Turkish texts (∼6GB). The corpus was preprocessed by first discarding non-Turkish content as identified

[4] Here, a refers to the ill-formed word, b refers to an IV word candidate, and x refers to any other candidate.

[5] Different strategies could be followed to compute normalization scores such as defining a precedence order over the kinds of similarities.

[6] http://www.kemik.yildiz.edu.tr/?id=28

by a language detection tool. Later, tweet-specific terms (e.g., # and RT), URLs, some punctuations, and repetitive characters were cleaned from the corpus. Finally, the remaining sentences of $\sim$9GB were tokenized into 7,401,321 distinct words.

We constructed a bipartite graph from the collected corpus using 5-gram word sequences. The word sequences contained one or more OOV words and in cases where there was only one OOV word, it might not be at the center. Statistics about the constructed graph is given in Table 1.

Bipartite	Number of Nodes	Average Degree	Average Edge Weight
Word	5,122,873	114.56	1.18
Context	567,078,012	1.03	

Table 1: Statistics about the graph.

In the study, if more than one initial node was determined for an ill-formed word, an equal number of random walks were performed from each of these nodes rather than selecting only one. 50 random walks with a maximum of 6 steps were performed from each initial node. Tailoring threshold was set to 1 for central contexts (a tailored context might have at most three $\emptyset$ symbols) whereas to 2 for sliding contexts. The edit distance threshold was set to 2 and contextually/positionally similar normalization candidates with a higher edit distance were discarded from consideration. Words were preprocessed before computing their edit distances. For instance, repetition of characters was reduced to a single character and words were deasciified to restore Turkish characters (if necessary). It is noteworthy to mention that our system do not normalize a word if an IV word candidate is not identified for that word.

Our test set consisted of 715 sentences that were retrieved from Twitter. The ill-formed words in these sentences were manually identified and normalized by two native Turkish speakers. 483 of these sentences contained only one ill-formed word. However, the remaining 232 sentences had at least two ill-formed words at different positions. Some sentences even had a number of ill-formed words in sequence. Our test set consisted of 1190 ill-formed words in total. Moreover, we observed different kinds of transformations between ill-formed words and their normalized forms (e.g., asciified Turkish characters, omitted characters, and typos).

As an example, consider one of our test sentences where ill-formed words are underlined "insanıar köprü kuracakıarı yerde duvar ördükıeri için yaınız kaıırıar"{People remain alone since they put up walls rather than building bridges}. The following shows how that sentence is normalized by a native speaker and by our system using different kinds of similarities:

Normalized Sentence: "insanlar köprü kuracakları yerde duvar ördükleri için yalnız kalırlar"
Contextual Similarity: "insanlar köprü kuracakları yerde duvar ördükleri için yalnız kalırlar
Positional Similarity: "insanıar köprü kuracakları yerde duvar ördükıeri için yaınız kaıırıar"
Lexical Similarity: "insanlar köprü kuracakları yerde duvar ördükleri için yalnız karar"

To our best knowledge, there is only one publicly available Turkish text normalization system (Torunoğlu and Eryiğit, 2014) that we can use in this evaluation in addition to a dictionary based MsWord spell checker. Neither the normalization system (TE_N) nor the spell checker (SP_C) utilizes the context of an OOV word during normalization. Table 2 shows the normalization results that we achieved on the whole test set along with the results of other normalizers. The third row (Con. Sim.) shows the performance of our system once only contextual similarities were considered during normalization ($\lambda_1 = 1, \lambda_2 = 0, \lambda_3 = 0$). Similarly, the fourth and fifth rows show the system's performance once positional and lexical similarities were considered respectively. The sixth row presents our baseline where all kinds of similarities had an equal effect on normalization ($\lambda_1 = 1, \lambda_2 = 1, \lambda_3 = 1$). The last row shows the results of our proposed system whose λ values were tuned on a development set of 200 sentences.

Normalizers	Precision	Recall	F-measure
SP_C	73.7%	71.5%	72.6%
TE_N	80.6%	77.3%	78.4%
Con. Sim.	73.6%	49.1%	58.9%
Pos. Sim.	88.1%	4.4%	8.3%
Lex. Sim.	86.1%	85.3%	85.7%
Con.+Pos.+Lex. Sim.	81.0%	80.3%	80.6%
Our System	**87.1%**	**86.3%**	**86.7%**

Table 2: Evaluation of all test sentences.

As shown in Table 2, a very low recall and the highest precision were obtained once the system considered only positional similarities between an OOV word and its IV word candidates. Moreover, our system achieved the highest F-measure and recall, and the second highest precision among all normalizers. Table 3 and Table 4 show the results obtained by our baseline and our system on sentences with only one OOV word and multiple OOV words respectively. Our system obtained higher scores in test sentences with multiple OOV words as compared to those with a single OOV word. This might be explained by the fact that a sentence written by the same person might contain multiple OOV words that underwent the same transformation (such as repetitive characters) which can be normalized by our system. On the other hand, several sentences might contain a single OOV word that experienced different kinds of transformations, some of which might not be handled by our system.

Normalizers	Precision	Recall	F-measure
Con.+Pos.+Lex. Sim.	79.3%	78.9%	79.1%
Our System	**81.9%**	**81.6%**	**81.8%**

Table 3: Evaluation of sentences with one OOV word.

Normalizers	Precision	Recall	F-measure
Con.+Pos.+Lex. Sim.	82.3%	81.3%	81.8%
Our System	**90.7%**	**89.7%**	**90.2%**

Table 4: Evaluation of sentences with multiple OOV words.

Overall, the evaluation results support our methodology for normalizing Turkish social media texts. It was our observation that replacing an OOV word with its normalized form before normalizing the next OOV word in the same sentence might end up with a totally different normalization. In the future, we will analyze the effects of such on-the-fly replacements on a large test set.

6 Conclusion

In this work, we describe a normalization system for social media texts. The system utilizes contextual and lexical similarities between non-standard and standard words for normalizing noisy texts. To encode contextual similarities, a bipartite graph is constructed from a large collection of texts where words and contexts that they appear in are connected with weighted edges. The graph is traversed via random walks in order to determine contextually similar normalization candidates for an ill-formed word. In determining the initial node(s) of these walks, input context of the ill-formed word is used either as is or after being tailored. An external word lexicon is used to explore normalization candidates that are lexically similar to the ill-formed word. The best candidate is selected based on its assigned similarity score. Our experiments on 715 Turkish tweets showed that our system achieved the state-of-the-art results. As future work, we plan to evaluate the system performance on other languages and to study how randomness in random walks can be reduced or totally eliminated to better gauge contextual similarities of words in real texts. Unfortunately, the morphological parser erroneously identifies some words (e.g., proper nouns) as OOV words and forces the system to treat these words as non-standard words. We will also study how this problem can be overcomed by using of other NLP tools such as named entity recognizers in the future.

References

AiTi Aw, Min Zhang, Juan Xiao, and Jian Su. 2006. A phrase-based statistical model for sms text normalization. In *Proceedings of the 21st International Conference on Computational Linguistics*, pages 33–40.

Eric Brill and Robert C. Moore. 2000. An improved error model for noisy channel spelling correction. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 286–293.

Çağıl Sönmez and Arzucan Özgür. 2014. A graph-based approach for contextual text normalization. In *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pages 313–324.

Monojit Choudhury, Rahul Saraf, Vijit Jain, Animesh Mukherjee, Sudeshna Sarkar, and Anupam Basu. 2007. Investigation and modeling of the structure of texting language. *International Journal on Document Analysis and Recognition*, 10:157–174.

Grzegorz Chrupala. 2014. Normalizing tweets with edit scripts and recurrent neural embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 680–686.

Paul Cook and Suzanne Stevenson. 2009. An unsupervised model for text message normalization. In *Proceedings of the 4th Workshop on Computational Ap-*

proaches to Linguistic Creativity (CALC), pages 71–78.

Jacob Eisenstein, Brendan O'Connor, Noah A. Smith, and Eric P. Xing. 2014. Diffusion of lexical change in social media. *PLoS ONE*, 9:71–76.

Jennifer Foster, Özlem Çetinoğlu, Joachim Wagner, Joseph Le Roux, Stephen Hogan, Joakim Nivre, Deirdre Hogan, and Josef van Genabith. 2011. # hardtoparse: Pos tagging and parsing the twitterverse. In *Proceedings of the Workshop On Analyzing Microtext (AAAI)*, pages 20–25.

Bo Han, Paul Cook, and Timothy Baldwin. 2013. Lexical normalization for social media text. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4:1–27.

Hany Hassan and Arul Menezes. 2013. Social text normalization using contextual graph random walks. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 1577–1586.

Amac Herdagdelen. 2013. Twitter n-gram corpus with demographic metadata. *Language Resources and Evaluation*, 47:1127–1147.

Max Kaufmann and Jugal Kalita. 2010. Syntactic normalization of twitter messages. In *Proceedings of the International conference on natural language processing*.

Dilek Kucuk and Ralf Steinberger. 2014. Experiments to improve named entity recognition on turkish tweets. In *Proceedings of the 5th Workshop on Language Analysis for Social Media*, pages 71–78.

Vivek Kumar and Rangarajan Sridhar. 2015. Unsupervised text normalization using distributed representations of words and phrases. In *Proceedings of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 8–16.

Fei Liu, Fuliang Weng, Bingqing Wang, and Yang Liu. 2011. Insertion, deletion, or substitution?: Normalizing text messages without pre-categorization nor supervision. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (HLT)*, pages 71–76.

Fei Liu, Fuliang Weng, and Xiao Jiang. 2012. A broad-coverage normalization system for social media language. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 1035–1044.

H. Andrew Schwartz, Johannes C. Eichstaedt, Margaret L. Kern, Lukasz Dziurzynski, Stephanie M. Ramones, Megha Agrawal, Achal Shah, Michal Kosinski, David Stillwell, Martin E. P. Seligman, and Lyle H. Ungar. 2013. Personality, gender, and age in the language of social media: The open-vocabulary approach. *PLoS ONE*, 8.

Katia Tautanova and Robert C. Moore. 2002. A pronunciation modeling for improved spelling correction. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 144–151.

Dilara Torunoğlu and Gülşen Eryiğit. 2014. A cascaded approach for social media text normalization of turkish. In *Proceedings of the 5th Workshop on Language Analysis for Social Media (LASM)*, pages 62–70.

Savaş Yıldırım and Tuğba Yıldız. 2015. An unsupervised text normalization architecture for turkish language. In *16th International Conference on Intelligent Text Processing and Computational Linguistics (CICLING)*.