TECNE: Knowledge Based Text Classification Using Network Embeddings

Rima Türker[1,2], Maria Koutraki[1,2], Lei Zhang[1], and Harald Sack[1,2]

[1] FIZ Karlsruhe – Leibniz Institute for Information Infrastructure, Germany
[2] Karlsruhe Institute of Technology, Institute AIFB, Germany
{firstname.lastname}@fiz-karlsruhe.de
{firstname.lastname}@kit.edu

Abstract. Text classification is an important and challenging task due to its application in various domains such as document organization and news filtering. Several supervised learning approaches have been proposed for text classification. However, most of them require a significant amount of training data. Manually labeling such data can be very time-consuming and costly. To overcome the problem of labeled data, we demonstrate TECNE, a knowledge-based text classification method using network embeddings. The proposed system does not require any labeled training data to classify an arbitrary text. Instead, it relies on the semantic similarity between entities appearing in a given text and a set of predefined categories to determine a category which the given document belongs to.

1 Introduction

Text classification is gaining more and more attention due to the availability of a huge number of text data, which includes search snippets, news data as well as text data generated in social networks. Recently, several supervised approaches have been proposed for text classification [6,1]. However, they all require a significant amount of labeled training data. Manual labeling of such data can be a very time-consuming and costly task. Especially, if the text to be labeled is of a specific scientific or technical domain, crowd-sourcing based labeling approaches do not work successfully and only expensive domain experts are able to fulfill the manual labeling task. Alternatively, semi-supervised text classification approaches [5] have been proposed to reduce the labeling effort. Yet, due to the diversity of the documents in many applications, generating small training set for the semi-supervised approaches still remains an expensive process [2]. Moreover, to cope with the problem of labeled data several *dataless text classification* methods have been proposed. Similar to our proposed approach, the methods do not require any labeled data, rather they rely on the semantic similarity between documents and the predefined categories. However, the most prominent and successful dataless classification approaches cannot utilize the rich entity and category information in large-scale knowledge bases.

In this paper we demonstrate TECNE, an approach which classifies an arbitrary input text, according to a predefine set of categories, without requiring

any training data. The approach is able to capture the semantic relation between the entities represented in a text and the predefined categories by embedding them into a common vector space using state-of-the-art network embedding techniques. Finally, the category of the given text can be derived based on the semantic similarity between entities (present in the given text) and a set of predefined categories. The similarity is computed based on the vector representation of the entities and the categories.

2 Description of TECNE

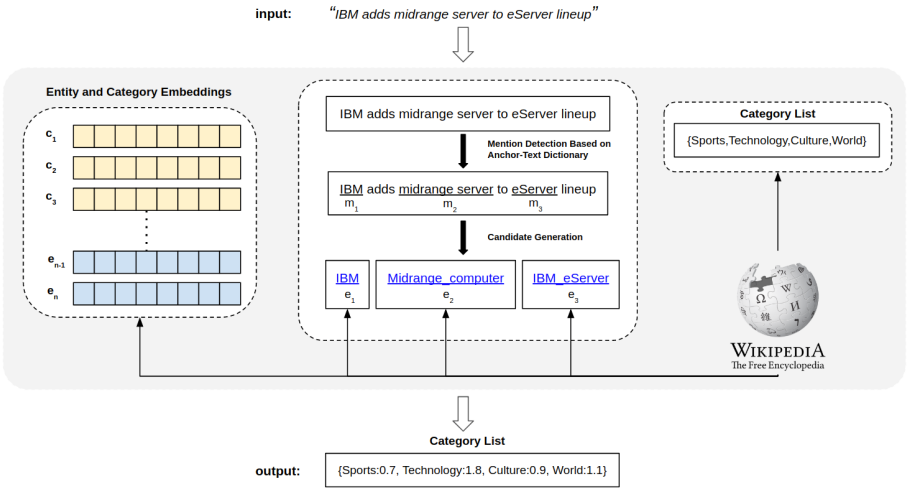

Fig. 1. The work flow of TECNE (best viewed in color)

Given a Knowledge Base KB, containing a set of entities $E = \{e_1, e_2, .., e_n\}$ and a set of hierarchically related categories $C = \{c_1, c_2, .., c_m\}$, where each entity $e_i \in E$ is associated with a set of categories $C' \subseteq C$ via a relation $cat \subseteq E \times C$, such that $cat(e_i) = C'$. The input of the system is an arbitrary text t, which contains a set of mentions $M_t = \{m_1, \ldots, m_k\}$ that uniquely refer to a set of entities as well as a set of predefined categories $C' \subseteq C$ (from the underlying knowledge base KB). The output of TECNE is a score value for each category $c_i \in C'$ based on the semantic similarity between the given text t and the predefined categories C'.

TECNE Overview The general work flow of TECNE presented in Figure 1 is similar to our previous study [4].
The input is a text t of an arbitrary length. Then, the classification task start with the detection of each entity mention present in t based on a prefabricated *"Anchor-Text Dictionary"* from Wikipedia. The Anchor-Text Dictionary contains all mentions and their corresponding Wikipedia entities. In our example the detected mentions are "IBM", "midrange computer" and "eServer".

As a next step, for each detected entity mention in t, the candidate entities are generated with the help of the Anchor-Text Dictionary. In our example

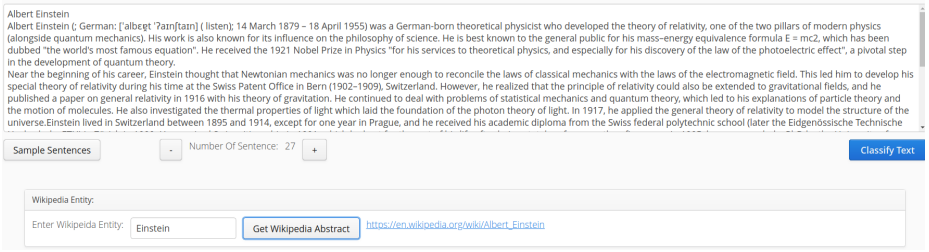

Fig. 2. Example of an input to the system using Wikipedia API

these are "IBM", "Midrange_computer" and "IBM_eServer". Also, the predefined categories (`Sports`, `Technology`, `Culture`, `World`) are mapped to Wikipedia categories. Finally, based on the entity and category embeddings [3] that have been precomputed from Wikipedia, the output of TECNE is a score for each predefined category. Ideally, the most semantically related category to the entities present in the input text should have the highest score. Thereby, in the given example the category *Technology* has the highest score. More technical details about the approach and the evaluation of the system can be found in [4].

3 Demonstration

A recorded video of our demonstration can be found here: https://goo.gl/pSxkcy TECNE is implemented in Java using a client-server architecture with communication over HTTP. The server is a RESTful web service, which is implemented using Spark[1]. Moreover, the client user interface is achieved by using Vaadin Framework[2] as a Web Application. The system supports both service-oriented and user-oriented interfaces for classifying short/long text documents. The system accepts any arbitrary text that needs to be classified as an input. In the interest of convince, the system utilizes three different APIs that a user can use to provide an online text as an input to the system. The first API[3] is used to fetch an abstract of Wikipedia articles. Simply, a user can enter the name of the Wikipedia article and the abstract of the certain article would be fetched automatically. Figure 2 presents the screen shot of this service, where the input is the abstract of the Albert Einstein's Wikipedia page. The second API[4] and the third API[5] are used to retrieve long and short random news respectively from different web pages. Besides that, a user can select a predefined sample sentence as an input or also manually enter any text without using the already provided data sources.

For the sake of simplicity, the system covers 4 different categories, i.e. the system can classify a text based on 4 different categories, `Sports`, `Business`, `World`, and `Science-Technology`. However, it can be easily extended to support higher number of categories for the classification purpose. For classifying a text, TECNE proceeds in 3 main steps as following:

[1] http://sparkjava.com [2] https://vaadin.com/ [3] https://en.wikipedia.org/w/api.php
[4] https://webhose.io/ [5] https://newsapi.org

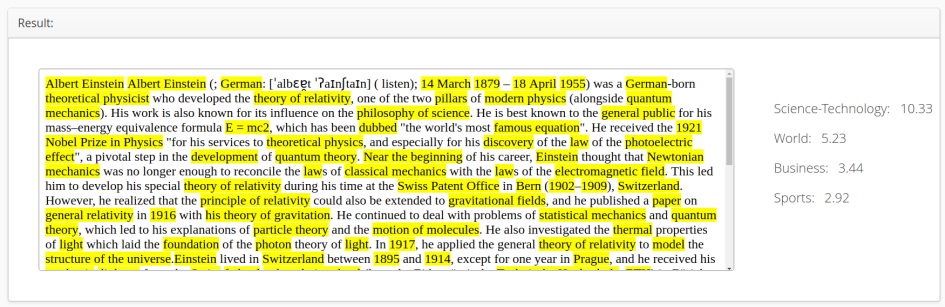

Fig. 3. Example of the detected mentions and the classification result

1 **Mention Detection Based on Anchor-Text Dictionary:** Each entity mention present in a given text is detected based on a *"Anchor-Text Dictionary"*. Figure 3 presents the screen shot of the detected mentions of the input article.
2 **Candidate Generation:** For each detected mention in the given input text, the candidate entities are generated based on the Anchor-Text Dictionary. For example, the first detected mention is "Albert Einstein", then the generated candidate entity is "Albert_Einstein"[6].
3 **Classification:** Finally, with the help of entity and category embeddings a score will be calculated for each category based on the assigned entities. Figure 3 presents an example of a classification result for the given text (abstract of the Albert Einstein's Wikipedia page). Based on the scores the categories are arranged in descending order.

4 Conclusion and Future Work

In this paper, we demonstrate TECNE, a system for knowledge based text classification using network embeddings. Future works also include the extension of TECNE towards enabling user to define a category list where the input text will be categorized accordingly.

References

1. Biswas, R., Türker, R., Moghaddam, F.B., Koutraki, M., Sack, H.: Wikipedia infobox type prediction using embeddings. In: DL4KGS@ESWC (2018)
2. Li, C., Xing, J., Sun, A., Ma, Z.: Effective document labeling with very few seed words: A topic model approach. In: CIKM. pp. 85–94. ACM (2016)
3. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: Line: Large-scale information network embedding. CoRR (2015)
4. Türker, R., Zhang, L., Koutraki, M., Sack, H.: "the less is more" for text classification. SEMANTICS (2018)
5. Xuan, J., Jiang, H., Ren, Z., Yan, J., Luo, Z.: Automatic bug triage using semi-supervised text classification. CoRR (2017)
6. Zhang, X., Zhao, J.J., LeCun, Y.: Character-level convolutional networks for text classification. In: NIPS (2015)

[6] https://en.wikipedia.org/wiki/Albert_Einstein