1 Data acquisition, preprocessing and classification algorithm

This section gives a high-level overview of the data acquisition and preprocessing workflow of textual content from firms' websites and of the design of a classification algorithm with the objective to learn firms' products and technologies. The steps we describe in the following are depicted in Figure (1).

First, for each company's domain[1], a web crawler[2] downloads all textual content and links by considering the content on the landing page and one page behind. For each webpage, the crawler creates a file that contains the textual content and the URL path of the respective domain.

Second, conditional on that a webpage's content is in English[3], we preprocess the crawled text data, i.e. text is tokenized, uncapitalized, stopwords and non-alphanumeric characters are removed. Then, words are mapped to the respective IDs of the word embeddings trained on data crawled from the web[4] that we use in the subsequent analysis (Mikolov et al., 2018). The usage of word embeddings allows us to capture the semantic similarity of the information on firms' webpages.

Third, since there exist several webpages of a company that (potentially) contain irrelevant information without capturing intrinsic, discriminatory details, we want to remove these webpages to obtain a more precise description of companies' products and technologies (Kinne and Lenz, 2019). We use simple heuristics to label a webpage as irrelevant, e.g. login pages, privacy statements or general disclaimers, or as relevant, e.g. webpages on products and services, by identifying keywords in a webpage's URL. This data sample is then used to train a binary classifier that labels a webpage either as relevant or irrelevant using a Convolutional Neural Network (CNN) that have been very successful particularly in computer vision and in natural language processing tasks. Figure (2) presents an exemplary schematic overview of the CNN architecture following Kim (2014) that we will employ for multiple classification tasks in steps 3 and 4. Additionally, we remove textual content from firms' webpages that appears on every webpage of a firm, e.g. social media links or menu bars.

Fourth, the final step is the main classification problem, i.e. we want to create a mapping from the text data of firms' webpages to their product and service (4.1-4.3 in Figure (1)) and technology (4.4) spaces using again a CNN architecture with slight modifications to the one exploited in step 3. We have described the creation of the input data in steps 1-3 and now differentiate between four classifications tasks with different target variables. On the one hand, with respect to firms' products and services, we will solve a multi-class classification problem where the output variable is a firm's primary North American Industry Classification System (NAICS)[5] code. We split the list of firms for that we have primary NAICS codes such that we have a stratified training and validation data set, fit the CNN model (incl. tuning of hyperparameters) and use the fitted model to classify unlabeled firms into a primary NAICS group. Additionally, we will formulate a multi-label classification problem with a firm's secondary NAICS code(s) as the output variable(s) and proceed analogous to the case of primary NAICS codes. Moreover, we will run a multi-label classification using more granular product and service descriptions of companies from Factset Revere[6] as output variable. This procedure allows one to obtain a complete overview of the market landscape regarding, for instance, new product markets and changes in competition over time and space.

[1] We obtain a list of companies and their corresponding webpages' URLs from Orbis.

[2] Our webcrawling algorithm is built on top of the open source webcrawling framework Scrapy, see https://scrapy.org/.

[3] An algorithm classifies the language of every textual data point (Joulin et al., 2016b,a). We use a pre-trained model for language identification, i.e. https://fasttext.cc/docs/en/language-identification.html.

[4] https://fasttext.cc/docs/en/english-vectors.html

[5] https://www.census.gov/eos/www/naics/

[6] https://open.factset.com/products/factset-rbics/en-us

On the other hand, concerning firms' technology portfolios, we will classify firms' patents based on their textual descriptions into the corresponding patent class(es) according to the International Patent Classification (IPC)[7]. This allows us to create a mapping from input text data to technology classes that we can then use to determine firms' technologies based on their websites' content. Thus, we transfer a learnt text-technology mapping from patent data to our setting of firms' information they disclose on their websites. By applying this method, we can determine the technology portfolios of firms that do not file patents and thus, obtain are more realistic depiction of the innovativeness of (small- and mid-sized) firms.

Figure 1: Data preprocessing and classification workflow.

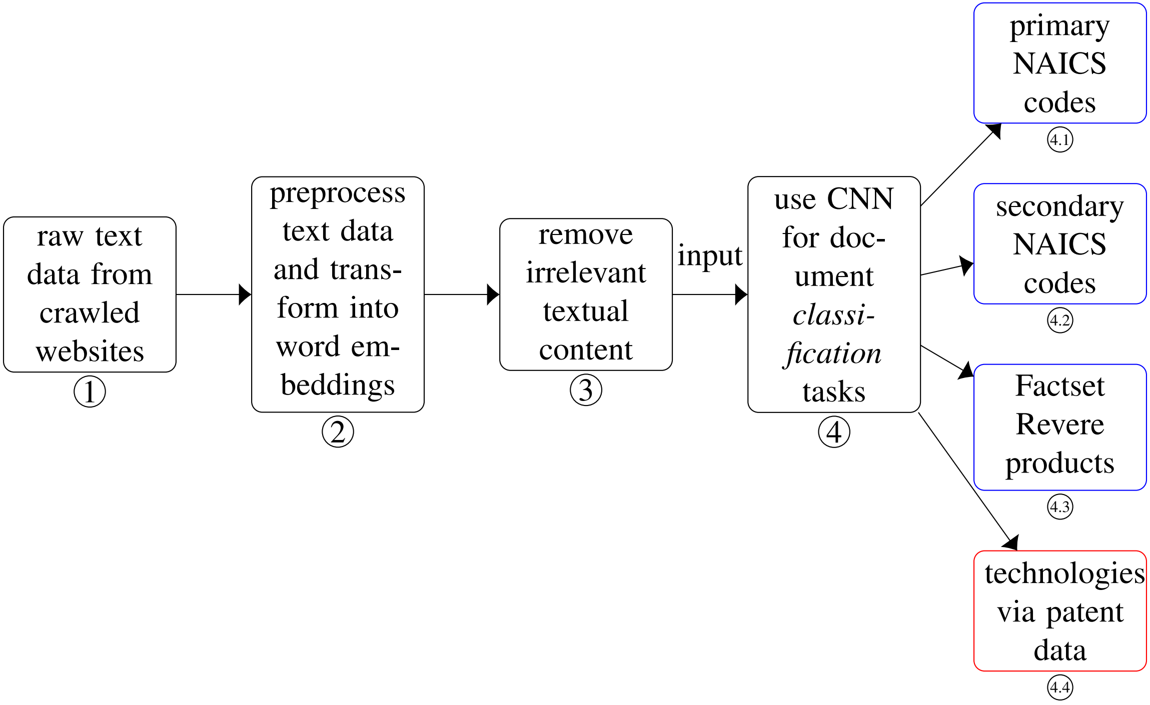
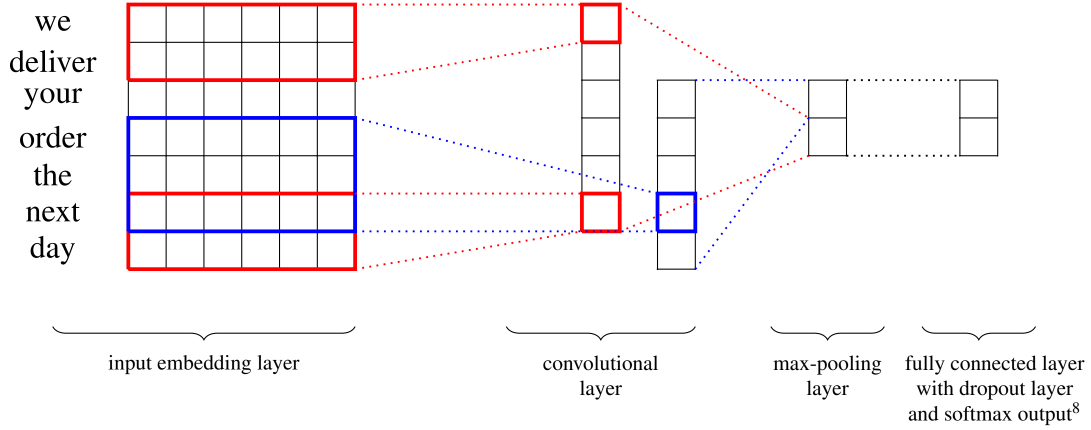

raw text data from crawled websites
①
preprocess text data and transform into word embeddings
②
remove irrelevant textual content
③
input
use CNN for document *classification* tasks
④
primary NAICS codes
4.1
secondary NAICS codes
4.2
Factset Revere products
4.3
technologies via patent data
4.4

Figure 2: CNN architecture for a binary classification task following Kim (2014) with two filters - of size of two (red) and three (blue) - and each with the number of filters set to one.



[7] https://www.wipo.int/classifications/ipc/en/

[8] We use a softmax function for multi-class classification problems and a sigmoid function for multi-label classification problems.

References

Joulin, A., Grave, E., Bojanowski, P., Douze, M., Jégou, H., and Mikolov, T. (2016a). Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2016b). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Kinne, J. and Lenz, D. (2019). Predicting innovative firms using web mining and deep learning. *ZEW Discussion Paper*, (19-001).

Mikolov, T., Grave, E., Bojanowski, P., Puhrsch, C., and Joulin, A. (2018). Advances in pre-training distributed word representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.