Ranking Sentences Describing Relationships Between Financial Entities by Relevance

Tim Repke
Hasso Plattner Institute
Potsdam, Germany
tim.repke@hpi.de

Michael Loster
Hasso Plattner Institute
Potsdam, Germany
michael.loster@hpi.de

Ralf Krestel
Hasso Plattner Institute
Potsdam, Germany
ralf.krestel@hpi.de

1 INTRODUCTION

Evaluating the credibility of a company is an important and complex task for financial experts. When estimating the risk associated with a potential asset, analysts rely on large amounts of data from a variety of different sources, such as newspapers, stock market trends, and bank statements. Finding relevant information in mostly unstructured data is a tedious task and examining all sources by hand quickly becomes infeasible.

An important aspect of risk management are the relations of a company of interest to other financial entities. Automatically extracting such relationships from unstructured text files, such as 10-K filings, significantly reduces the amount of manual work. Such structured knowledge enables experts to quickly gain insight into a company's relationship network. However, not all extracted relationships may be important in a given context. In this paper, we propose an approach to rank extracted relationships based on text snippets, such that important information can be displayed more prominently.

2 DATASET

The dataset used for this work was provided in the context of the FEIII Challenge 2017[1], which contains almost 1000 *triples* extracted from 25 10-K and 10-Q filings, which describe a relationship (*role*) between the *filing company* and a *mentioned financial entity*. The context a relation is appeared in the original filing, is given by text snippets of three sentences. Relationships are limited to ten predefined roles (see table 1). Judging from their respective text snippets, triples were labelled by experts according to their relevance from a business perspective as *irrelevant*, *neutral*, *relevant*, or *highly relevant*.

Task Description. The challenge is aimed to explore methods that automatically produce a ranking by relevance of triples with the same role. This complements last year's challenge to identify financial entities in free text[1].

Inter Annotator Agreement. The quality of the annotations is estimated using Cohen's Kappa[2] ($\kappa \in [0, 1]$), which quantifies the inter annotator agreement (IAA) between two experts as shown in figure 1. Around 40% of the triples were rated by more than one expert with a weighted average of $\overline{\kappa} = 0.45$, which indicates a high level of disagreement. For our training and evaluation purposes, we map the ratings to numerical values (1–4) and consider the discrete average rating for each triple labelled by multiple experts.

3 OUR APPROACH

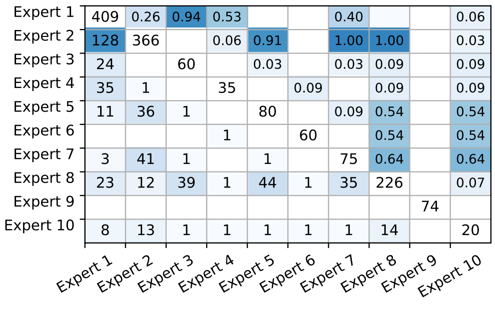
We rank the snippets for each role based on an ensemble of multi-class classifiers. The ensemble consists of three multi-class classifiers, all trained on different feature sets. Two classifier consist of four one-versus-rest logistic regression models trained on the experts' labels. The one-versus-rest logistic regression output probabilities of all four classifiers are fed into a softmax function to calculate a weighted ranking score. The third classifier is a random forest model. The different models of the ensemble are based on different feature sets, namely *bag-of-words* (BOW), *embeddings* (EMB), and *syntax features* (SYN).

3.1 Bag-of-Words

Our first model uses a simple bag-of-words representation of the snippets to classify them. N-grams are extracted for $n = 1$ to 3 and are weighted based on information gain between the classes. In order to reduce the feature space and guard against over-fitting, a filter removes the most and least frequent terms.

3.2 Sentence Embeddings

Difficulties with previously unseen examples might arise from the limited training size. Word embeddings can alleviate this problem by representing words in a 50- to 300-dimensional vector space. These representations are learned by using unsupervised deep learning.



	Expert 1	Expert 2	Expert 3	Expert 4	Expert 5	Expert 6	Expert 7	Expert 8	Expert 9	Expert 10
Expert 1	409	0.26	0.94	0.53			0.40			0.06
Expert 2	128	366		0.06	0.91		1.00	1.00		0.03
Expert 3	24		60		0.03		0.03	0.09		0.09
Expert 4	35	1		35		0.09		0.09		0.09
Expert 5	11	36	1		80		0.09	0.54		0.54
Expert 6				1		60		0.54		0.54
Expert 7	3	41	1		1		75	0.64		0.64
Expert 8	23	12	39	1	44	1	35	226		0.07
Expert 9									74	
Expert 10	8	13	1	1	1	1	1	14		20

Figure 1: κ Inter Annotator Agreement (upper triangular matrix), number of commonly rated triples (lower triangular matrix), and number of ratings (diagonal matrix)

[1] https://ir.nist.gov/feiii/

DSMM'17, Chicago, USA

123-4567-24-567/08/06...$15.00
DOI: 10.475/123_4

Table 1: Averaged experimental results for each role using BOW

	affiliate	agent	counterpart	guarantor	insurer	issuer	seller	servicer	trustee	underwriter
# samples	185	61	64	34	19	129	20	21	420	21
NDCG (5-fold-cv)	0.97	0.97	0.93	0.97	0.99	0.92	1.0	0.98	0.99	1.0
Baseline (random)	0.86	0.81	0.84	0.96	0.97	0.75	0.96	0.93	0.88	0.96

Table 2: Experimental results for bag-of-words (BOW), embedding (EMB), syntax (SYN) features, and ensemble

Approach	NDCG	σ(NDCG)	F1-Score	σ(F1)
Baseline (random)	0.87	0.07	-	-
Baseline (worst)	0.73	0.13	-	-
BOW	**0.98**	0.03	0.73	0.27
EMB	0.92	0.07	0.41	0.16
SYN	0.94	0.06	0.43	0.26
BOW+EMB+SYN	0.93	0.06	0.46	0.22

Internally, a neural network is trained to predict the following word in a sequence of words based on the word's context window.

We learned paragraph embeddings[2] from 25 of the original full text filing documents containing 60k sentences (2m words). Previous research has shown, that such embeddings manage to outperform BOW approaches[3]. We use a window size of 10 and a paragraph vector of size 50 (approximately three sentences), which is trained for 30 epochs over all filings. From the embedding, we induce a vector for each sentence in the text snippet associated with a triple and concatenate them.

3.3 Syntax Features

Additionally, to provide a language independent approach, we created a set of syntax-based features. Following the Gini impurity metric, features, such as the ratio of upper-case words and numbers, or the number of dollar signs and word repetitions, appear to be most meaningful for classification. In total we derived 25 features describing the number or presence of different syntactical characteristics.

In our experiments, a logit model has proven to be a good choice for BOW and EMB, but has shown unsatisfactory performances on syntax features. Therefore, we chose random forests, which perform much better in this case.

3.4 Ensemble

Each of the numerical representations and their resulting models have their strengths and weaknesses. For example, the language independence of SYN can tolerate a changing vocabulary to a certain extent, but misses the advantage to identify key phrases which may prove useful for classification. As a conclusion, we combined the three models by summing the individual predictions to form a soft vote.

4 EVALUATION

The system's performance is measured by normalised discounted cumulative gain (NDCG)[2]. We perform 5-fold cross-validation (5-fold-cv). Table 2 lists the mean NDCG scores and the standard deviation (σ). For comparison, we consider a baseline of the worst possible ranking (inverse order of the ideal ranking) and the average of multiple random rankings. The BOW model performed best in our experiments, the EMB and SYN models show similar results. Training a model on text usually requires a reasonably large corpus. With the data at hand we had to pay close attention to the feature selection, since seemingly very specific terms are likely to negatively affect the model's ability to classify unseen samples. Looking at the performance of the classification task itself (measured by the F1-Score) the EMB model has the smallest standard deviation across multiple runs of the evaluation. Combining different text representations did not produce an improved model on this limited set of data. The soft-voted ensemble of all classifiers shows similar results as EMB and SYN alone.

5 CONCLUSION

Overall, we managed to achieve very good NDCG scores of around 0.98 using a BOW model. However, with the limited amount of data at hand, concerns about the ability to generalise to unseen samples that deviate from the vocabulary can not be fully eliminated. We assume EMB may be more robust to such changes, since the underlying word embeddings are trained on a significantly larger set of text and is able to reflect phrase similarities. A comparison of the standard deviation of the F1-Scores of 0.27 and 0.17 respectively, supports this assumption.

For future work we are interested in additional external data, e.g. the impact of a business relationship may be judged by comparing revenues of the involved companies. Thus, triples could be enriched by adding (historical) revenue of the two involved financial entities.

REFERENCES

[1] 2016. *DSMM'16: Proceedings of the Second International Workshop on Data Science for Macro-Modeling.* ACM, New York, NY, USA.

[2] Bruce Croft, Donald Metzler, and Trevor Strohman. 2009. *Search Engines: Information Retrieval in Practice.* Pearson.

[3] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Conference on Neural Information Processing Systems 2013.*

[2] Using Gensim https://radimrehurek.com/gensim/models/doc2vec.html