Text Classification of Patents Using the K-Nearest Neighbor Algorithm

Catelyn Scholl and John W. Sheppard
Gianforte School of Computing, Montana State University

Abstract

TechLink is partnered with the Department of Defense (DoD) to transfer DoD technology to industry. Due to the high number of patents registered with the US Patent and Trademark Office, their objective is to find an efficient way to identify DoD technology. To reduce expert labor, we used a machine learning approach to classify patents. We introduce text classification using the bag-of-words and tf-idf models with the K-Nearest Neighbor algorithm to determine if a patent is owned by the DoD.

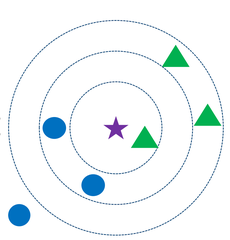
Sample DoD Patent Title
Selectable lethality yield inflatable grenade
Sample Non-DoD Patent Title
See-through computer display systems

Introduction

K-Nearest Neighbor (KNN) is a simple yet effective machine learning algorithm. KNN finds the *k* closest training points (vectors) according to a distance metric. The predicted class is then determined by the majority vote.

Advantages:

- Easy to understand
- Conceptually easy to implement
- General approach / adaptable



Limitations:

- Sensitive to noise
- Computationally expensive
- "Curse of dimensionality" (Kozma, 2008)

Methodology

Data processed through TechLink filters

CSV file: application ID, title and label

Ground truth shuffled and split into 80% train and 20% test using sklearn train_test_split

Dictionary Representation					
[Docs, Terms]	network	compute	system	graph	army
D1	4	5	1	0	0
D2	0	0	3	0	4
D3	0	0	0	2	0
D4	0	1	3	0	0

Unigram Bag of Words (BoW) – Count Vectorizer

Term Frequency Inverse Document Frequency - TF-IDF Vectorizer

Jaccard:
$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

Cosine:
$$\cos(\theta) = \frac{A \cdot B}{\| A \| \| B \|}$$

SciKit-Learn KNN Classifier

Results

k	Model	Distance Metric	Word Root	Accuracy (%)
1	BoW	Jaccard	Lem	74.572
1	BoW	Jaccard	Stem	73.817
1	TF-IDF	Cosine	Lem	75.356
1	TF-IDF	Cosine	Stem	75.297
5	BoW	Jaccard	Lem	73.788
5	BoW	Jaccard	Stem	74.224
5	TF-IDF	Cosine	Lem	78.055
5	TF-IDF	Cosine	Stem	78.462
10	BoW	Jaccard	Lem	72.308
10	BoW	Jaccard	Stem	73.179
10	TF-IDF	Cosine	Lem	79.448
10	TFIDF	Cosine	Stem	78.781

Algorithms used to compute nearest neighbors include brute force, ball tree, and KD tree (Pedregosa, F, et al., 2011).

References

Kozma, L. (2008). *k Nearest Neighbors algorithm (kNN).* Helsinki University of Technology.

Pedregosa, F, et al. (2011). Scikit-learn: Machine Learning in Python. *JMLR, 12*, pp. 2825-2830.

Acknowledgements

Special thanks to NSF, Dr. John Sheppard, Amy Peerlinck, and Na'Shea Wiesner.