Identification, Tracking and Impact: Understanding the Trade Secret of Catchphrases

Jagriti Jalal  
Dept. of Computer Science and Engg.  
IIT Kharagpur, India  
jagritijalal@iitkgp.ac.in

Mayank Singh  
Dept. of Computer Science and Engg.  
IIT Gandhinagar, India  
singh.mayank@iitgn.ac.in

Arindam Pal  
Data61, CSIRO, Cyber Security CRC  
Sydney, NSW, Australia  
arindamp@gmail.com

Lipika Dey  
TCS Innovation Labs, India  
lipika.dey@tcs.com

Animesh Mukherjee  
Dept. of Computer Science and Engg.  
IIT Kharagpur, India  
animeshm@cse.iitkgp.ac.in

1 INTRODUCTION

As software and other products are becoming more complex, the number and size of patent documents are increasing gradually. Automated patent document processing systems are essential to extract information and gain insights from this ever-increasing collection of patent databases. Catchphrases provide a concise representation of the content of a document. A catchphrase is a well-known word or phrase encapsulating the particular concept or subject of a document. They contain all the important legal and technical aspects, instead of just summarizing the document. They have numerous applications such as document categorization, clustering, summarization, indexing, topic search, quantifying semantic similarity with other documents, and conceptualizing particular knowledge domain of the document [12, 16]. However, since only a small minority of documents have author-assigned catchphrases, and manual assignment of catchphrases to existing documents is time-consuming, the automation of the catchphrase extraction process is highly desirable. In the current study, catchphrases represent innovation topics. Figure 1 presents example catchphrases from two different patent abstracts.

In this paper, we propose an unsupervised method for the extraction of catchphrases from the abstracts of patents granted by the U.S. Patent and Trademark Office over the years. The key contributions of this paper are as follows.

• We propose an unsupervised technique for catchphrase identification and ranking in patent documents.
• We conduct robust evaluations and comparison against several state-of-the-art baselines.
• As a secondary objective, we study the evolution of catchphrases present in the patents filed by various organizations over time.
• We bring forth some of the unique temporal characteristics of these catchphrases and show how these are correlated to the overall future citation count of the patents filed by an organization.
• The catchphrase evolution study further unfolds that companies get polarized based on whether the patent documents keep re-using the same catchphrases over time or they introduce newer catchphrases as time progresses.
Figure 1: Example abstracts from USPTO patents US06681004 and US06680003. The highlighted set of words are identified as catchphrases from IPC (described in Section 5).

2 RELATED WORK

A variety of techniques have been applied for automated keyword extraction like locating important phrases by analyzing markups like capitalization, section headings and emphasized texts [17]; building phrase dictionary by parts-of-speech (POS) tagging of word sequences [18]; thesaurus-based keyphrase indexing [29]; domain-specific keyphrase extraction [11, 23] and several other supervised methods such as KEA [30], MAUI [22], back-of-the-book indexing using catchphrase extraction [10], MAUI with text denoising [26], CSSeer [8] etc. In recent years artificial neural networks (ANNs) are being used to build predictive models to rank words in a document [5] and then select keywords based on these ranks. It has been widely recognized that the innovative capability of a firm is a critical determinant of its performance and competitive edge [3, 13, 15]. Since patents are a direct outcome of the inventive process and are broken down by technical fields, they are considered indicators of not only the rate of the innovative activities of a firm but also its direction [1, 2, 4]. Many previous studies have examined the relationships between the patenting activities of a company and its market value [14, 24]. Bornmann and Daniel [6] precisely reviews the citing behavior of scientists and shows the role of citations as a reliable measure of impact. Cheng et al. [9] shows that some indicators of patent quality are statistically significant to return on assets. Lee et al. [19] assesses future technological impacts by employing the future citation count as a proxy while Lee et al. [20] employs various patent indicators such as novelty and scope, as features of an ANN for early identification of emerging technologies.

3 DATASETS AND PREPROCESSING

The current study requires a rich time-stamped dataset. We, therefore, leverage two independent data sources. These are:

1. The patent dataset: We compile the first dataset by crawling the full-text patent articles, available at the United States Patent and Trademark Office (USPTO\(^1\)). It comprises patents granted weekly (Tuesdays) from January 1, 2003, to May 18, 2018 (excluding images/drawings). The patents are available as XML encoded files with English as the primary language. Out of all the curated documents, in this study, we only consider those patents for which the abstract information is present (see Table 1 for statistics).

2. The newsgroup corpus: We also use another data source, the 20 Newsgroups Dataset\(^2\) donated by T. Mitchell in 1999. It includes one thousand Usenet articles each from 20 newsgroups like ‘alt.atheism’, ‘comp.graphics’, ‘talk.politics.guns’, etc. Approximately 4% of the articles are crossposted. This serves as a non-patent corpus to estimate the importance of a word specifically in the domain of the patents concerning a non-patent domain (see Table 1 for statistics).

| Patent ID | Year range | Number of patents | Number of patents with abstract |
|-----------|------------|-------------------|--------------------------------|
| US06681004 | 2003–2018  | 3,915,639         | 3,486,866                     |

| Newsgroup ID | Year range | Number of articles | Number of words | Language |
|--------------|------------|--------------------|-----------------|----------|
| US06680003   | 1993–2017  | 19,997             | ~               | English  |

Table 1: General statistics about the patent dataset and the newsgroup corpus. A large fraction (89%) of patents have abstract information.

Pre-processing: For both of the above, we performed several pre-processing tasks such as a sentence to lowercase conversion, removal of special characters except apostrophe and periods, lemmatization, and multiple white-spaces removals.

4 CATCHPHRASE EXTRACTION

Catchphrase extraction is a challenging problem mainly due to the diversity and unavailability of large-text annotated datasets. We, therefore, present an unsupervised method for catchphrase extraction. We propose a two-stage extraction strategy that identifies relevant candidate catchphrases in a given patent article. In the first stage, we select the candidate catchphrases. This is followed by candidate catchphrase ranking in the second stage. Next, we describe the two stages in detail.

\(^1\)https://bulkdata.uspto.gov/
\(^2\)https://archive.ics.uci.edu/ml/datasets/Twenty+Newsgroups
### 4.1 Stage-1: Candidate selection

In the first stage, we select candidate catchphrases from each patent’s abstract. Empirically, we observe that all catchphrases are n-gram noun phrases, for example, unigrams (e.g., `communication`, `dielectrometry`, etc.), bigrams (e.g., `consecutive bit`, `voice synthesizer`, etc.), trigrams (e.g., `integrated circuit device`, `hydrogen chloride gas`, etc.) or quadrigrams (e.g., `commercially available synthesis tool`, `electric signal processing board`, etc.). We, therefore, perform part-of-speech-tagging (POS) of each abstract text to identify noun phrases. Currently, we leverage python’s state-of-the-art NLP library SpaCy\(^3\). Note that, we experimented with two text processing approaches before noun phrase identification: (i) with stopwords (WS), and (ii) without stopwords (WOS). WS represents that no stop-words were removed from the abstracts, whereas, WOS represents that all stop-words in the abstract text were removed beforehand. Abstracts with stopwords (WS) led to better quality extraction results due to the existence of stop-words in noun phrases. We discuss the results in detail in Section 5. Table 2 presents statistics of extracted candidate phrases from the dataset.

| Word n-grams | Count |
|--------------|-------|
| Unigrams     | 208,105 |
| Bigrams      | 2,616,762 |
| Trigrams     | 4,432,251 |
| Quadrigrams  | 2,138,696 |
| Total        | 9,395,814 |

Table 2: Count of n-gram noun phrases generated from patent dataset.

### 4.2 Stage-2: Candidate ranking

Candidate phrases obtained in the first stage are ranked in this stage. The ranking algorithm is based upon two empirical findings: (i) how well the phrase describes the document’s topic, and (ii) how specific is the phrase to the patent literature. Our proposed method unifies both of these findings by combining a frequency-based measure with an information-theoretic measure. Given a patent document \(d\) and a set of candidate phrases \(c_d\) obtained in the previous stage, we compute the phrase score \(PS(c, d)\) for each phrase \(c \subseteq c_d\).

\[
PS(c, d) = \sum |c| \left( \log(score(t_i)) \right) \cdot KLI(c, d) \tag{1}
\]

where, \(t_i\) denotes the \(i^{th}\) term in an n-gram candidate phrase \(c\), \(score(t_i)\) denotes the score of the \(i^{th}\) term by estimating the importance of the term specifically in the patent domain relative to a non-patent domain and \(KLI(c, d)\) represents the Kullback-Leibler divergence informativeness specifying how well a candidate phrase \(c\) represents a document \(d\). The term score \(score(t)\) in the above equation is computed as

\[
\text{score}(t) = \frac{Importance(t, C_p)}{Importance(t, C_n) + 1} \tag{2}
\]

Again, here, \(C_p\) and \(C_n\) represents the patent collection and non-patent (in our case, the newsgroup) collection. The importance \((t, C)\) of a term \(t\) in a given collection \(C \in \{C_p, C_n\}\) is measured in terms of the collection frequency \(CF\) and the document frequency \(DF\). \(CF(t, C)\) represents how many times the term \(t\) appeared in the entire collection \(C\). \(DF(t, C)\) represents the count of documents where the term \(t\) appeared. It is computed as

\[
Importance(t, C) = \frac{CF(t, C)}{DF(t, C) + 1} \tag{3}
\]

\(KLI(c, d)\) denotes an information theoretic measure to compute how informative the phrase is in the given document \(d\). It is computed as:

\[
KLI(c, d) = \frac{TF(c, d)}{|d|} \cdot \log \frac{TF(c, d)}{CF(c)} \tag{4}
\]

where, \(TF(c, d)\) represents how many times \(c\) appeared in document \(d\). \(CF(c)\) denotes how many times \(c\) appeared in the entire patent collection \(C_p\). \(|d|\) and \(|n|\) represents total number of n-grams in document \(d\) and \(C_p\) respectively.

The above scoring method results in a ranking of candidate phrases. We select top-ranked candidates such as top-5, top-10, top-20, etc., and evaluate our unsupervised method in the next section.

## 5 EXPERIMENTS

In this section, we describe the experimental settings, baselines and the evaluation metrics. We construct a collection of possible catchphrases from the International Patent Classification (IPC) list. This list is maintained by the World Intellectual Property Organization (WIPO)\(^4\). The IPC provides a hierarchical system of language independent symbols for the classification of patents and utility models according to the different areas of technology to which they pertain. The hierarchy comprises eight high-level categories:

1. **Cat-1**: Human necessities
2. **Cat-2**: Performing operations; Transporting
3. **Cat-3**: Chemistry; Metallurgy
4. **Cat-4**: Textiles; Paper
5. **Cat-5**: Fixed constructions
6. **Cat-6**: Mechanical engineering; Lighting; Heating; Weapons; Blasting
7. **Cat-7**: Physics
8. **Cat-8**: Electricity

In each of these high-level categories, several sub-categories exist. An n-gram phrase represents each category. We term these phrases as `ground truth catchphrases` (GTC). Overall, we obtained 22,855 GTC such as "actuators", "cleaning fabrics", "feedback arrangements in control systems", etc. We use GTC to evaluate our proposed catchphrase extraction method. Table 3 presents examples of GTC for each high-level category. Next, we present three state-of-the-art baselines.

### 5.1 Baselines

1. **Keyphrase extraction algorithm (KEA):** KEA [30] is a supervised machine learning toolkit that extracts keyphrases

\(^3\)https://spacy.io/usage/linguistic-features

\(^4\)http://www.wipo.int/classifications/ipc/ipcpub/
and ranks them. The original algorithm was trained on scientific documents and uses a trained Naïve Bayes model. We trained KEA for patent documents leveraging a similar training procedure.

(2) Legal: Mandal et al. [21] also follow an unsupervised approach for identification of catchphrases from legal court cases. The scoring is done as:

\[
P_S(c, C_p, C_{np}) = \log \left[ \sum_{i=1}^{|c|} \text{Score}(t_i, C_p, C_{np}) \right] \cdot \text{KL}(c, d) \quad (5)
\]

where score\((t_i, C_p, C_{np})\) and KL\((c, d)\) can be calculated using equations 2 and 4 respectively. Note the change in the formula in equation 5 compared to equation 1. This modification as we shall see almost doubles our performance.

(3) KLIP: Tomokio and Hurst [27], Verberne et al. [28] proposed a Kullback-Leibler (KL) divergence based phrase assignment score which is a linear combination of two different scores:

- KL informativeness (KLI): KLI measures how well the candidate phrase represents a document. It is computed using equation 4.
- KL phraseness (KLP): KLP score is computed specifically for multi-word phrases. It compensates for low frequency of multi-word phrases by assigning higher weights to longer phrases:

\[
\text{KLP}(c, d) = \frac{\text{TF}(c, d)}{|d|} \cdot \log \frac{\text{TF}(c, d)}{\prod_{i=1}^{\text{freq}(t_i, d)} |d_i|} \quad (6)
\]

where, \(t_i\) is the \(i\)th term of the phrase \(c\), and \(\text{freq}(t_i, d)\) is the frequency of the term \(t_i\) in document \(d\).

(4) BM25: BM25 [25] is a well-known measure for scoring documents with respect to a given query. We use this function for assigning score to an extracted candidate phrase \(c\) in a given document \(d\). The scoring function is:

\[
\text{score}(c, d) = \text{IDF}(c) \cdot \frac{\text{TF}(c, d)(k_1 + 1)}{\text{TF}(c, d) + k_1 \left( 1 + b \frac{|d|}{\text{avg}d} \right)} \quad (7)
\]

where \(\text{TF}(c, d)\) is the term frequency of phrase \(c\) in the document \(d\), \(k_1\) and \(b\) are free parameters. We choose \(k_1 \in [1, 2.0]\) and \(b = 0.75\). \(\text{IDF}(c)\) is the inverse document frequency of the candidate phrase \(c\), calculated as

\[
\text{IDF}(c) = \log\left( \frac{n - DF(c) + 0.5}{DF(c) + 0.5} \right) \quad (8)
\]

where \(DF(c)\) is the document frequency of the phrase \(c\) in the collection.

Note that KEA is a supervised machine learning model whereas Legal, KLIP and BM25 are unsupervised methods.

### 5.2 Evaluation measures

We evaluate our proposed method against the three baselines. We use two standard evaluation measures: (i) **Macro precision**, and (ii) **Macro recall**. These metrics are computed by macro-averaging the precision/recall values computed for every patent.

\[
\text{Macro precision} = \frac{\sum_{i=1}^{|T|} \text{precision}_i}{|T|} \quad (9)
\]

\[
\text{Macro recall} = \frac{\sum_{i=1}^{|T|} \text{recall}_i}{|T|} \quad (10)
\]

where \(\text{precision}_i\) and \(\text{recall}_i\) are the precision and recall values computed for the \(i\)th patent in our test dataset \(T\). The precision and recall values for the \(i\)th patent are computed as follows

\[
\text{precision}_i = \frac{\text{DCP}_i}{\text{DC}_i} \quad (11)
\]

\[
\text{recall}_i = \frac{\text{DCP}_i}{\text{CPC}_i} \quad (12)
\]

where \(\text{DC}_i\), \(\text{DCP}_i\), and \(\text{CPC}_i\) represents the number of catchphrases in the \(i\)th patent that are detected, detected and present in GTC, and present in GTC respectively.

As KEA requires training, we partition our dataset into two classes: (i) train and (ii) test. Train split consist of 2,055,588 (65%) patent documents. Test split consist of 1,106,883 (35%) patent documents. For a fair comparison, we evaluate our proposed method against baselines (described in Section 5.1) using only the test split.

### 5.3 Results and discussion

Table 4 compares our proposed catchphrase extraction approach against state-of-the-art baselines. We outperform all baselines by a substantially high margin. The second best system in terms of precision is KEA, whereas the second-best system in terms of recall is a mix between KLIP and KEA. The baseline Legal performed
worst among all the baselines, which is possible because of the fact that the authors take a logarithm of the sum of all the scores rather than the sum of the logarithms of the scores. The former measure undermines the contribution of the scores from each term and is therefore ineffective and is rather unintuitive.

6 TEMORAL STUDY

In this section, we intend to show the usability of catchphrase extraction. We claim that catchphrase evolution presents a fair understanding of the changing innovation trends of companies. We conduct several interesting temporal studies to understand the emergence of new research topics in the industry. In this study, we select top-10 companies from three industrial segments: (1) Software, (2) Hardware, and (3) Mobile Phones. Table 5 presents the list of top-10 companies in each of the above three segments.

In subsequent sections, we analyze patents filed by these companies over the years. In our patent dataset, each company can have several variations in name due to multiple research groups, geographical locations, subsidiaries, headquarters, etc. For example, IBM is present as ‘International Business Machines Corporation Armonk’, ‘International Business Machines Laboratory Inc.’, etc. We overcome these inconsistencies by manually annotating name variations. However, we claim that basic string matching techniques can easily automate this normalization. Besides, we eliminate frequently occurring catchphrases like, ‘method’, ‘present invention’, etc., to ignore noisy/redundant signals. This filtering process was automated by removing catchphrases with top-10 document frequencies. We next, present how catchphrases can be leveraged in understanding the topical evolution of companies.

6.1 Topic evolution

In this section, we study the topical evolution of companies. We leverage the Jaccard Similarity (JS) between the catchphrases to compute the topical overlap between patents filed in consecutive years by a specific company. We conduct this experiment for 11 years between 2006–2016. Figure 2 shows temporal profiles of a three-year moving average over JS for each of the three segments. We observe that Baidu in Hardware segment while Oppo, Vivo, and OnePlus in Mobile Phones segments exhibit relatively low similarity between catchphrases over the years. However, most of the companies have similarity curves with multiple peaks with an overall increase in the JS values over the years. For this analysis, we only considered 2-gram catchphrases. However, we found similar observations for higher n-gram catchphrases. If an organization is filing patents on the same topics over the years, the JS value will only increase; on the other hand, if an organization is continuously filing patents on newer topics, the JS value is expected to decline.

6.2 Categorization

Further, we conduct a nuanced study to understand this temporal behavior. We classify each company’s similarity profile into five categories [7] based on the number and location of peaks. A peak in the similarity profile of a company represents a high topical similarity between consecutive years followed by a topical drifting off period. We leverage the peak identification method proposed by Chakraborty et al. [7]. Note that peaks occurring in consecutive years are considered as a single peak. The categories are:

1. MonInc: Similarity profile that monotonically increases. The peak occurs in the last year.
2. MonDec: Similarity profile that monotonically decreases. The peak occurs in the first year.
3. PeakInit: Similarity profile that consists single peak within the first three years but not the last year.
4. PeakLate: Similarity profile that consists single peak after the initial three years but not the last year.
5. PeakMult: Similarity profile consisting of multiple peaks.
6. Others: Similarity profiles that do not qualify into the above categories are kept in this category. They mainly consist of profiles with extremely low JS values for each year.

Table 6 shows categorization results. We find no company in MonInc and PeakInit categories. Majority of the companies are present in the PeakMult category followed by PeakLate category. Companies in Others category have very less number of filed patents. Three out of four companies in Others category are recently launched mobile companies.

Even though, PeakMult category consists multiple peaks, we observe two distinct fluctuation patterns. We term these patterns as (i) stable and (ii) unstable. In stable, the profile looks considerably less fluctuating. The profile highly fluctuates in unstable category. We quantify the above fluctuating patterns by leveraging the average value of JS. Given, JS(c) is the similarity profile for a company c, average value of JS (avgJS(c)) is computed as:

\[
\text{avgJS}(c) = \frac{\min(JS(c)) + \max(JS(c))}{2}
\]

Empirically, we observe that companies with avgJS > 0.1 can be classified as unstable, while the rest can be classified as stable. Table 7 shows companies in the PeakMult category that are further categorized into stable and unstable. Among, stable and unstable sub-categories, the former contains more (=7) companies than the latter (=5).

6.3 Citation count

Citations, in the scholarly world, determine the popularity of research papers/authors/organizations. Here, we adopt a similar analogy for patent articles. A patent citation is a document cited by an applicant, third party, or a patent office examiner because its content relates to a patent application. We compute the citation count of a patent p by summing the citations received by p. For the current study, we construct citer-cited pairs by extracting references present in patent texts and use these pairs to compute patent citation counts.

Next, we create multiple citation zones based on the citation count of a patent. We define four distinctive zones: (i) very low, (ii) low, (iii) medium, and (iv) high, to study the influence of the JS profile of a company on the number of citations received by its patents. Table 8 presents zoning statistics of the complete dataset. Out of 3,829,153 patent articles, 1,499,175 have zero citation count.
Table 4: Comparison of our proposed method against the baselines: Precision and recall values at different top-ranks ($z \in 10, 15, 20, 15\%, 20\%$) of extracted catchphrases.

| $z$ | Our Model | KEA | Legal | BM25 | KLIP | Our Model | KEA | Legal | BM25 | KLIP |
|-----|-----------|-----|-------|------|------|-----------|-----|-------|------|------|------|-------|------|-------|------|-------|------|-------|------|-------|------|-------|------|-------|------|-------|
| 10  | 0.253     | 0.192| 0.075 | 0.120|      | 0.773     | 0.155| 0.255 | 0.120|      |
| 15  | 0.231     | 0.148| 0.128 | 0.131| 0.146| 0.910     | 0.155| 0.146 | 0.131| 0.146|
| 20  | 0.217     | 0.133| 0.156 | 0.156| 0.164| 0.945     | 0.155| 0.164 | 0.156| 0.164|
| 15\%| 0.260     | 0.200| 0.056 | 0.060| 0.108| 0.750     | 0.155| 0.056 | 0.060| 0.108|
| 20\%| 0.240     | 0.156| 0.109 | 0.111| 0.132| 0.886     | 0.155| 0.109 | 0.111| 0.132|

Figure 2: Moving average of catchphrase similarity between consecutive years for – Software (left), Hardware (center), and Mobile Phone (right) companies.

Table 5: Top-10 Software, Hardware, and Mobile Phone companies selected from three publicly available lists.

| Category | Count | Names |
|----------|-------|-------|
| MonInc   | 4     | Tencent, Samsung, Xiaomi, Lenovo |
| MonDec   | 0     |       |
| PeakInit | 0     |       |
| PeakLate | 6     | Facebook, TCS, Huawei, AsusTek, Foxconn, Compal |
| PeakMult | 12    | HP, SAP, Accenture, Nokia, Fujitsu, Quanta, Quanta Computer, Microsoft, IBM, Oracle, Google, Apple, LG |
| Others   | 4     | Baidu, Oppo, Vivo, OnePlus |

Table 6: Categorization of top-10 Software, Hardware, and Mobile Phone companies based on temporal catchphrase similarity profile. No company was classified in MonDec and PeakInit category.

Next, we relate similarity profiles and citation count zones. For each company, we measure the fraction of patents in different citation zones. We leverage histograms as a visualization tool to
conduct this study. In Figure 3, we observe that the fraction of patents in Medium and High citation zones in PeakLate category are relatively higher than in MonInc category. This indicates that the introduction of diversity in topics over time helps in enhancing the future citations of the patents filed by a company.

Figure 3: Citation count zones vs similarity profiles: Fraction of patents in PeakLate (left) and MonInc (right) category companies in each citation count zone.

While the stable group is concerned more about computer systems, the unstable group is more about electronic device parts.

6.5 Temporal visualizations

In this section, we study the catchphrase evolution of companies. As a popular visualization tool, we leverage word clouds. We create word clouds for each company between the years 2003–2016. Due to space constraints, in Figure 6, we only consider word clouds for one representative company from stable, unstable, peaklate and moninc categories at three representative years. We claim that catchphrase evolution presents a fair understanding of the changing innovation trends of companies. Note that we consider only bigram catchphrases in this study. We can conduct a similar study for any company in different years.

In Figure 6a, we study catchphrase evolution for Microsoft (a representative company in the stable group). We observe a shift from traditional topics such as client-server models, databases, basic Web development, etc. (in 2003), toward full-fledged Web search and Internet technologies (in 2010). In 2016, the focus shifted to mobile devices and gesture identification. The above trends coincide with several product releases such as BING (a search engine released in 2009) and Lumia (mobile phones released in 2015).

In Figure 6b, we study catchphrase evolution for Oracle (a representative company in the unstable category). Oracle seems to have shifted its focus from traditional database topics like relational databases, query, etc. (in 2003), toward the development of software as a service (SAAS) in 2010. In 2016, it continued to focus on services with a major emphasis on reliable authentication mechanisms in the cloud. These innovation trends resulted in several products like Oracle cloud (cloud computing service launched in 2016), Primavera (an enterprise project portfolio management software acquired by Oracle in 2008), etc.

Similarly, in Figure 6c, we study catchphrase evolution for Facebook (a representative company in PeakLate category). As Facebook started its operations from 2004, we present visualizations...
Figure 6: Word clouds of the representative companies in different similarity profile based categories at three distinct years. (a) stable (Microsoft), (b) unstable (Oracle), (c) peaklate (Facebook), and (d) moninc (Samsung).
for three years, 2010, 2013 and 2016. The initial focus was to develop technical features like news feeds, membership, etc. In the year 2013, these trends shift toward instant messaging aspects. In the year 2016, the catchphrases show a distinct innovation pattern of restricting and disclosing data availability. Facebook Messenger (introduced in 2011) is one of the products developed between 2011–201312.

12https://en.wikipedia.org/wiki/Facebook

We study Samsung as a representative company in MoNcs category (see Figure 6d). Primarily Samsung’s major focus lies in traditional electronics innovation. Recent trends suggest an increased focus on mobile technologies such as user interfaces, display units, etc.
7 CONCLUSION AND FUTURE WORK

In this paper, we propose an unsupervised catchphrase identification and ranking system. Our proposed system achieves a substantial improvement, both in terms of precision and recall, against state-of-the-art techniques. We demonstrate the usability of this extraction by analyzing how topics evolve in patent documents and how these evolution patterns shape the future citation count of the patents filed by a company.

In the future, we plan to extend the current work by developing an online interface for automatic catchphrase identification. We also plan to study the influence of catchphrase evolution on the company’s revenue.

REFERENCES

[1] Daniele Archibugi and Mario Planta. 1996. Measuring technological change through patents and innovation surveys. Technovation 16, 9 (1996), 543–551.

[2] Kendall W. Artz, Patricia M. Norman, Donald E. Hatfield, and Laura B. Cardinal. 2010. A longitudinal study of the impact of R&D, patents, and product innovation on firm performance. Journal of product innovation management 27, 5 (2010), 725–740.

[3] Richard A. Bettis and Michael A. Hitt. 1995. The new competitive landscape. Strategic management journal 16, S1 (1995), 7–19.

[4] Nicholas Bloom and John Van Reenen. 2002. Patents, real options and firm performance. The Economic Journal 112, 478 (2002), C97–C116.

[5] Zvi Boger, Tsvi Kuflik, Peretah Sheval, and Bracha Shapira. 2001. Automatic keyword identification by artificial neural networks compared to manual identification by users of filtering systems. Information Processing & Management 37, 2 (2001), 187–198.

[6] Lutz Bornmann and Hans-Dieter Daniel. 2008. What do citation counts measure? A review of studies on citing behavior. Journal of documentation 64, 5 (2008), 1–20.

[7] Tammik Chakraborthy, Suhansanu Kumar, Pawan Goyal, Niloy Ganguly, and Andries Mukherjee. 2015. On the categorization of scientific citation profiles in computer science. Commun. ACM 58, 9 (2015), 82–90.

[8] Hung-Hsuan Chen, Pucktada Treeratpituk, Prasenjit Mitra, and C. Lee Giles. 2013. CSSeer: an expert recommendation system based on CiteSeerX. Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries, ACM, 381–382.

[9] Yin-Hui Cheng, Fu-Yung Kuan, Shih-Chieh Chuang, and Yun Ken. 2009. Profitability decided by patent quality? An empirical study of the US semiconductor industry. Scientometrics 82, 1 (2009), 175–183.

[10] Andras Csomai and Rada Mihalcea. 2008. Linguistically motivated features for enhanced back-of-the-book indexing. Proceedings of ACL-08: HLT (2008), 932–940.

[11] Eibe Frank, Gordon W. Paynter, Ian H Witten, Carl Gutwin, and Craig G. Neville-Manning. 1999. Domain-specific keyphrase extraction. In 16th International joint conference on artificial intelligence (IJCAI’99), Vol. 2. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 668–673.

[12] Parthasarathy Gopavaparthy, Line C Pouchard, and Santiago Pujol. 2016. Increasing datasets discoverability in an engineering data platform using keyword extraction. In Proceedings of the 16th ACM/IEEE-CS Joint Conference on Digital Libraries. ACM, 225–226.

[13] Christine Greenhalgh and Mark Longland. 2005. Running to stand still? the value of R&D, patents and trade marks in innovating manufacturing firms. International Journal of the Economics of Business 12, 3 (2005), 307–328.

[14] Bronwyn H Hall, Adam Jaffe, and Manuel Trajtenberg. 2005. Market value and patent citations. RAND Journal of economics (2005), 16–38.

[15] Constance E. Helfat and Margaret A. Peteraf. 2003. The dynamic resource-based view: Capability lifecycles. Strategic management journal 24, 10 (2003), 997–1010.

[16] Steve Jones and Gordon Paynter. 1999. Topic-based browsing within a digital library using keyphrases. In Proceedings of the fourth ACM conference on Digital libraries. ACM, 114–121.

[17] Bruce Kruhl and Chad Burkey. 1997. The InfoFinder agent: Learning user interests through heuristic phrase extraction. IEEE Expert 12, 5 (1997), 22–27.

[18] Leah S Larkey. 1999. A patent search and classification system. In Proceedings of the fourth ACM conference on Digital libraries. ACM, 179–187.

[19] Changyong Lee, Yangrae Cho, Hyonju Seol, and Yongtae Park. 2012. A stochastic patent citation analysis approach to assessing future technological impacts. Technological Forecasting and Social Change 79 (1) (2012), 16–29.

[20] Changyong Lee, Ohjung Kwon, Myeongjung Kim, and Daisel Kwon. 2015. Early identification of emerging technologies: A machine learning approach using multiple patent indicators. Technological Forecasting and Social Change 127 (2015), 291–306.

[21] Animesh Mukherjee. 2015. On the categorization of scientific citation profiles in computer science. Journal of the Economics of Business 76, 3 (2015), 307–328.

[22] Christine E. Helfat and Margaret A. Peteraf. 2003. The dynamic resource-based view: Capability lifecycles. Strategic management journal 24, 10 (2003), 997–1010.

[23] Steve Jones and Gordon Paynter. 1999. Topic-based browsing within a digital library using keyphrases. In Proceedings of the fourth ACM conference on Digital libraries. ACM, 114–121.

[24] Sooyoung Oh, Zhen Lei, Prasenjit Mitra, and John Yen. 2012. Evaluating and ranking patents using weighted citations. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. ACM, 281–284.

[25] Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: BM25 and beyond. In Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries. ACM, 281–284.

[26] Changyong Lee, Ohjung Kwon, Myeongjung Kim, and Daisel Kwon. 2015. Early identification of emerging technologies: A machine learning approach using multiple patent indicators. Technological Forecasting and Social Change 127 (2015), 291–306.

[27] Animesh Mukherjee. 2015. On the categorization of scientific citation profiles in computer science. Journal of the Economics of Business 76, 3 (2015), 307–328.

[28] Christine Greenhalgh and Mark Longland. 2005. Running to stand still? the value of R&D, patents and trade marks in innovating manufacturing firms. International Journal of the Economics of Business 12, 3 (2005), 307–328.