Abstract—There is an ever more rapidly increasing number of design and CAD conference and journal papers. There is also a large number of design researchers (paper authors) and a large number of microelectronics conferences. Significant government and industrial funding and even higher research effort is allocated to a great variety of topics. While some researchers keep changing collaborators, others maintain steady collaborations. The return on financial and human investment is sometimes spectacular but often minimal or none. Our conceptual aim is to help microelectronics researchers, students, designers, and managers better organize their professional activities. Our practical objective is to address some of the essential productivity questions for microelectronics researchers, including what papers to read, where to publish, and with whom to collaborate. Our scientific goal is to develop statistically sound technique that guide actions of highly successful microelectronics researchers, which we define for the purpose of this paper as ones that are the authors of the most cited papers. We show that one can surprisingly accurately predict the number of citations for a paper at the time of its publishing by considering factors such as citation factors of the authors and the quality of a conference or a journal. Interestingly, highly cited coauthors rarely improve the citation record of their collaborators.

Keywords—Citation Prediction; Machine Learning; Bibliometrics

I. INTRODUCTION

A. Motivation

Essential organizational and educational questions for any microelectronics researcher include which papers to read, where to publish, and with whom to collaborate. Unfortunately, the number of options alone is so high that it is often difficult to meaningfully answer these questions without the use of statistical and optimization techniques.

More than two hundred CAD and design papers are published every week. There are more than 50 design and integrated circuits magazines; there are thousands of design blogs and several mailing lists and forums. Microelectronics and CAD patents are among the most popular topics in all leading patent offices. The number of papers in related fields such as compilers and device technology is even much higher. For example, there were more than 8,000 carbon nanotubes papers published last year and more than 3,000 graphene papers in addition to 1,700 carbon nanotubes and 400 graphene issued patents [10]. Even the most enthusiastic microelectronics researcher can read only a small percentage of relevant publications.

According to Microsoft Academic Search there are 10,869 researchers with at least 20 citations in reputed publications. The number of citations is drastically different for different authors. It is tempting to think that coauthoring papers with the most cited authors or most prolific authors is very helpful for one’s own citation records. Furthermore, it is unclear whether it is better to publish at highly specialized smaller conference dedicated to an emerging topic or the target the most prestigious and established venue. In summary, each research must make a number of important decisions that impact not just her citations, but also the quality of her research.

B. Problem Formulation and Key Solution Concepts

At the core of our approach is analysis and prediction of the expected number of citations for a microelectronics paper at the moment of its publication. Essentially, for the purpose of our analysis we assume that the goal of each researcher is to publish highly cited papers. Of course, that is not completely correct assumption. However, we postulate that there is strong correlation between the number of citation and the actual impact of the paper. The correlation is even stronger, we assume, between the authors of highly cited papers and their impact. In addition, one line of reasoning is that one can benefit by reading papers that will be highly cited just after their publishing. Therefore, our goal is to predict the number of citations a given paper will receive over a specified period of time, as well as the likelihood that it will receive that number of citations, by using only data that is available at the time of publishing. This information includes several citation indices, the average number of citations for papers at given conferences/journals, and the citation data for coauthors.

There are several technical challenges that we must address, including:

- **High percentage of low cited papers.** Almost all papers have very few citations. Therefore, any algorithm is very accurate for a majority of papers if it predicts that any paper will have very few citations. The resolution between well performing algorithms is blurred due to the distribution of citations followed by power law.
- **Selection and/or development of classification algorithms.** There is a huge number of widely used classification procedures. It is tempting to develop a new classification technique. However, it is evident that we are in an area of diminishing returns. Our analysis shows that different classification algorithms best perform on different data sets. Therefore, we decided to look for better predictions elsewhere, i.e. in identifying best predictors.

**Identification of essential predictors.** Our procedure for finding accurate prediction parameters follows two dimensions. The first one
is that we are looking for predictors that are accurate by themselves. The second dimension is to identify principal components, i.e.,
predictors that predict well they are used simultaneously.

**Extraction of essential knowledge from prediction models.**

Finally, and maybe most importantly, one of our objectives is to extract causal relationship between each prediction parameter and the number of citations. This knowledge may be more important than the actual prediction of the number of citations for a given paper.

II. FRAMEWORK

We build a generic framework for citation prediction. The framework is fed with publication network data set as inputs, and terminates when the prediction accuracy cannot be improved. During each iteration, it selects one machine learning algorithm with its parameters to obtain the best prediction results as outputs.

![Block diagram of our citation prediction framework.](image)

**Problem Formulation**

For a given paper \( p \in P \), the goal is to predict citation count \( C_{py} \) at the year \( y \in Y \) after publication, where \( 1 \leq y \leq 10 \). We use only information available at publication time (zero citation history) as predictive features \( f \in F_d \) for individual data set \( d \in D \), extracted from the feature extraction scheme.

**Our Approach**

Followed by the feature extraction, we then sort a set of features \( F_d \) by one of two feature selection schemes \( e \in E \). We only take the top \( n \) features from the feature set \( F_e \), where \( n \) is dynamically determined depending on prediction accuracy.

We solve this citation prediction problem \( A \) as a multi-classification problem using a set of thresholds \( t \in T_d \), where \( t \) can be obtained from the distribution of citation counts in the data set \( d \). A training set \( P = \{ p_1, p_2, ..., p_l \} \) is a set of pre-classified samples. Each sample as a paper instance \( p \) consists of a vector \( [f_1, f_2, ..., f_n] \), where \( f_i \) denotes a feature of the sample. The training set is augmented with a vector \( C = [c_0, c_1, ..., c_q] \), where \( c_q \) represents the class to which each sample belongs. Using a classifier with multi-classes, we get citation prediction results to which class the predicted citations belongs. Prediction accuracy is defined as:

\[
prediction\_accuracy = \frac{N_c}{N_i}
\]

where \( N_c \) is the number of correctly classified instances, and \( N_i \) is the total number of instances.

Our goal is that no matter what data sets are used as inputs, the framework is automatically to extract a set of predictive features from the data set, build a classifier with multi-classes using a set of thresholds determined from the distribution of citations, select one ML algorithm to predict future citations, select the top \( n \) features by one of the ranking algorithms, and tune parameters of the ML algorithm to obtain the best results at the end. The framework depicted in Figure 1 runs until it obtains one best-performing ML algorithm with proper parameter settings.

III. DATA SET

The goal of data collection process is critical, since 24 bibliometric features listed in Table 1 are generated and used in prediction of citations with the help of Microsoft Academic Search which is an alternative to Google Scholar.

We select Hardware and Architecture (HW) as a primary field of study. This field will then be compared with the data set in Databases (DB). Firstly, these fields are two of the most active research fields in Computer Science within the past 10 years. As a result, we can obtain enough citation history from them. Prediction models are trained on articles from 1990 to 1995, while articles from 1996 to 1999 were excluded and used as a test set.

One interesting finding is that the distributions of citations both in conferences (CFs) and in journals (JRs) are pretty similar and roughly follow Newman’s power laws [2] as shown in Figure 2. The number of citations having a certain population size in terms of paper instances is found to vary as a power of the population size, and hence follows a power law:

\[
p(x) = Cx^{-\alpha}
\]

where \( \alpha \) is called the exponent of the power law with the requirement of \( \sum p(x) = 1 \). Once \( \alpha = 2.5 \) in [2]) is fixed, the constant \( C \) can be determined by

\[
C = (\alpha - 1)x^{\alpha-1}
\]

**Figure 2. Distribution of citations of papers published in conferences and journals in Hardware and Architecture.**

IV. MACHINE LEARNING ALGORITHMS

We consider seven Machine Learning (ML) algorithms to predict citations. C4.5, SVM, Logistic Regression (LR), and Bayesian Networks (BNs) are popular and heavily utilized in citation prediction literature while RIPPER (RIP), MultiLayer Perceptrons (MP), and k-Nearest Neighbor (kNN) are rarely explored. The in-depth details of each algorithm are out of scope of this paper. The process of parameter tuning is omitted for the page limitation.

V. FEATURES

The main goal for feature extraction and feature selection is to obtain a set of features that are able to predict future citation counts in an effective way.

A. Feature Extraction

There are 24 bibliometric features in \( F_d \) extracted from individual data set as shown in Table 1. For completeness, we introduce commonly accepted definitions of \( h \)-index, \( g \)-index, and \( e \)-index.

**Definition 1 (h-index).** A scientist has index \( h \) if \( h \) of his or her \( N_p \) papers have at least \( h \) citations each, and the other \( (N_p-h) \) papers have at most \( h \) citations each.

\( N_p \) denotes the number of papers published over \( n \) years. For \( h \)-index used in \( f_3, f_7, f_14, \) and \( f_25 \), a person with an index of \( h \) has published \( h \) papers, each of which has been cited by others at least \( h \) times [3].
Definition 2 (g-index). Given a set of articles ranked in decreasing order of the number of citations that they received, the g-index is the (unique) largest number such that the top g articles received (together) at least \( g^2 \) citations.

The g-index used in \( f_{18} \), \( f_{15} \), and \( f_{23} \) is very similar to the h-index, and attempts to address its shortcomings [4]. Recently, Zhang proposed a notion of e-index [5] that can complement the h-index for excess citations as used in \( f_{10} \), \( f_{13} \), and \( f_{24} \).

Definition 3 (e-index). The excess citations received by all papers in the h-core are denoted by \( e^j \):

\[
e^j = \sum_{i=1}^{h} (c_{ij} - h) = \sum_{i=1}^{h} c_{ij} - h^2
\]

where \( c_{ij} \) denotes citation counts for a paper j. We also consider the impact factor as \( f_{14} \). It is calculated by the number of citations gained from individual CF or JR \( f_{13} \) divided by the number of articles published from individual CF or JR \( f_{18} \) [5]. Paper counts are used in \( f_{13} \), \( f_{14} \), \( f_{15} \), and \( f_{19} \). Citations are used in \( f_{11} \), \( f_{12} \), and \( f_{16} \). Author counts are used in \( f_{17} \) and \( f_{18} \). Thus, we place more emphasis on how predictive the features are rather than content-based features.

### Table 1. List of features as inputs of prediction model training.

<table>
<thead>
<tr>
<th>ID</th>
<th>Type</th>
<th>Feature Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>f1</td>
<td>Number of authors for a given paper</td>
<td>Number of authors for a given paper</td>
</tr>
<tr>
<td>f2</td>
<td>Number of citations for all authors</td>
<td>Number of citations for all authors</td>
</tr>
<tr>
<td>f3</td>
<td>Average number of citations for all authors</td>
<td>Average number of citations for all authors</td>
</tr>
<tr>
<td>f4</td>
<td>Number of citations for first author</td>
<td>Number of citations for first author</td>
</tr>
<tr>
<td>f5</td>
<td>Number of papers written by first author</td>
<td>Number of papers written by first author</td>
</tr>
<tr>
<td>f6</td>
<td>Number of papers written by all authors</td>
<td>Number of papers written by all authors</td>
</tr>
<tr>
<td>f7</td>
<td>Average number of papers written by all authors</td>
<td>Average number of papers written by all authors</td>
</tr>
<tr>
<td>f8</td>
<td>h-index for first author</td>
<td>h-index for first author</td>
</tr>
<tr>
<td>f9</td>
<td>g-index for first author</td>
<td>g-index for first author</td>
</tr>
<tr>
<td>f10</td>
<td>e-index for first author</td>
<td>e-index for first author</td>
</tr>
<tr>
<td>f11</td>
<td>Sum of h-index for all authors</td>
<td>Sum of h-index for all authors</td>
</tr>
<tr>
<td>f12</td>
<td>Sum of g-index for all authors</td>
<td>Sum of g-index for all authors</td>
</tr>
<tr>
<td>f13</td>
<td>Sum of e-index for all authors</td>
<td>Sum of e-index for all authors</td>
</tr>
<tr>
<td>f14</td>
<td>Average h-index for all authors</td>
<td>Average h-index for all authors</td>
</tr>
<tr>
<td>f15</td>
<td>Average g-index for all authors</td>
<td>Average g-index for all authors</td>
</tr>
<tr>
<td>f16</td>
<td>Average e-index for all authors</td>
<td>Average e-index for all authors</td>
</tr>
<tr>
<td>f17</td>
<td>Number of co-authors for all authors</td>
<td>Number of co-authors for all authors</td>
</tr>
<tr>
<td>f18</td>
<td>Number of publication years for all authors</td>
<td>Number of publication years for all authors</td>
</tr>
<tr>
<td>f19</td>
<td>h-index for the most cited author</td>
<td>h-index for the most cited author</td>
</tr>
<tr>
<td>f20</td>
<td>g-index for the most cited author</td>
<td>g-index for the most cited author</td>
</tr>
<tr>
<td>f21</td>
<td>e-index for the most cited author</td>
<td>e-index for the most cited author</td>
</tr>
<tr>
<td>f22</td>
<td>Information for authors</td>
<td>Information for authors</td>
</tr>
<tr>
<td>fn</td>
<td>Feasibility Analysis</td>
<td>Feasibility Analysis</td>
</tr>
<tr>
<td>f23</td>
<td>Number of papers for conferences or journals</td>
<td>Number of papers for conferences or journals</td>
</tr>
<tr>
<td>f24</td>
<td>Impact factor for conferences or journals</td>
<td>Impact factor for conferences or journals</td>
</tr>
</tbody>
</table>

### Feature Selection

We consider two feature selection schemes that can rank features: regression coefficients (RC) and correlation coefficients (CC). The regression coefficient-based feature selection scheme is heavily utilized in the literature for feature selection [6]. However, our results in Section 6.B will show that the scheme is more unstable in terms of prediction accuracy than a CC-based approach as the feature set size increases. As an alternative approach, we denote the linear CC denoted by the quantity \( c \) that can be interpreted as the strength and the direction of a linear relationship between two features:

\[
c(f_j,f_i) = \frac{n \sum_{k=1}^{n} f_j f_k - \left( \sum_{k=1}^{n} f_j \right) \left( \sum_{k=1}^{n} f_k \right)}{\sqrt{\left( n \sum_{k=1}^{n} f_j^2 - \left( \sum_{k=1}^{n} f_j \right)^2 \right) \left( n \sum_{k=1}^{n} f_k^2 - \left( \sum_{k=1}^{n} f_k \right)^2 \right)}
\]

where \( 0 \leq |c| \leq 1 \). The feature \( f_j \in F_{20} \) is the citation counts at the target prediction year, and the predictive feature \( f_i \in F_{24} \). Features in Table 1 are later ranked by \( |c| \).

### Feature Pre-classification

Recently, Fu et al. developed a prediction model with a threshold-based binary classifier rather than continuous ones [6]. They argue that error metrics for continuous loss functions such as mean square error and absolute error are difficult to interpret when it comes to their practical significance. In our framework, we first adopt their approach, and then extend it to the classifier with multiple classes based on four thresholds denoted by \( T_1 \) to \( T_4 \). Unlike in [3], for fair allocation we set thresholds \( T_1 \), \( T_2 \), \( T_3 \), and \( T_4 \) to be citations gained at 80\%, 90\%, 95\%, and 98\% percentile of the distribution from the data set, respectively. The publication year of an individual paper is set to be 0. Citations are accumulative year after year.

### VI. RESULTS

We choose Weka [7] as a primary ML toolkit because it is open-source and has been widely used over 10 years with little concerns of implementation issues. We predict future citations for even years between 2 and 10 after publication. We extract a set of features that contain 24 bibliometric features, and sort them by correlation coefficients. By the feature selection scheme, the framework selects features from the sorted list. We then build prediction models on the training set and verify it on the testing set. Although the feature selection scheme can dynamically select the top \( n \) features for any arbitrary \( n \), it is reasonable to set \( n = 10 \).

#### A. Feasibility Analysis

To study feasibility of our approach, we first calculate correlation coefficients defined in Equation 4 using HW data sets. Figure 3(a-b) presents the intensity of correlations between future citation counts and each of features obtained at publication time. For journal papers, \( f_{18} \) and \( f_{23} \) can be thought of as the best predictive feature while \( f_{24} \) is not shown to be strong in conference papers. This difference is observed due to the degree of skewness on distribution of citations between two data sets. Overall, the number of citations from authors and that of published JRs/CFs are closely correlated with the future citations. In general, correlations are decaying as the year of prediction increases from 2 to 10. While the g-index in \( f_{15} \), \( f_{12} \), and \( f_{13} \) is weakly correlated, both in JRs and in CFs they can be thought of as relatively important predictive features than others such as \( f_1 \) to \( f_{12} \). Interestingly, the g-index for the most cited author for a given paper \( f_{12} \) shows relatively strong correlation. Thus, we note that h-index and e-index may be less important in citation prediction compared to the g-index. Features from conference papers are less correlated with future citations than ones from journal papers. Therefore, we expect to obtain more accurate prediction results with more correlated features.

#### B. Prediction Results

Figure 3(c-d) demonstrates prediction results by applying six different ML algorithms supported by Weka to our data set. The underlying assumption of our framework is that one ML algorithm cannot outperform over all data sets. Two plots also confirm that one winner in HW CFs may not become a winner on HW JRs. This rule holds even within the same data set as the prediction year differs. For example, LR is the best predictor for the 4\textsuperscript{th} year of prediction while C4.5 outperforms for the 10\textsuperscript{th} year in HW JRs. The advantage of using our approach is that no matter what data sets and ML algorithms are used, the framework can effectively extract a set of predictive features from the raw data to be applied in prediction.

Although correlation coefficients calculated are less than .5 as in Figure 3(b), the combination of top \( n \) features performs well with 82.5\% of accuracy in 10\textsuperscript{th} year prediction. We also applied it to the data set from the Database field, thus, generating 83.5\% of prediction accuracy in the same setting. Interestingly, in DB field the e-index used in \( f_{10} \), \( f_{13} \), \( f_{16} \), and \( f_{24} \) turns out to be a very important factor. This is because more papers in DB present excess citations ignored by h-index but captured by e-index as discussed in [5].
C. Sensitivity Analysis

We compare the CC-based approach with the regression coefficient (RC) based approach by analyzing how sensitive the prediction accuracy is as the number of features varies. Figure 3(c) presents the RC-based prediction to study how effective the selected features are. SVM, MP, RIP, and LR are less sensitive to the varied number of features used. However, bottom three ML algorithms such as C4.5, BNs, and kNN fluctuate, when increasing the feature set size with below 70% of prediction accuracy on average.

On the other hand, Figure 3(f) demonstrates the CC-based scheme. Roughly speaking, as more features are added to the training set, prediction accuracy slightly decreases. MP, SVM, RIP, LR, and C4.5 are pretty much static when adding more features in model training with above 80% of prediction accuracy on average. kNN and BNs however are more sensitive to the increase of feature set size. Compared to the RC scheme, this CC scheme outperforms in terms of stable prediction results as the number of features increases.

VII. RELATED WORKS

Despite the rise of citation analysis in various fields, few have attempted to study algorithms for predicting future citations due to the fundamental difficulty in predicting citations [1]. Researchers in [9] attempted to predict the total number of citations that papers acquire over a lifetime of 40 years. Recently, [8] studied a predictive regression model for predicting citations 2 years after publication using clinical articles. The state-of-the-art work in citation prediction [6] studied three prediction models, SVM, Logistic Regression, and Decision Tree using data sets from biomedical publications within a horizon of 10 years only predictive information available at publication time. Their work is different from our work in that we attempt to predict future citations with only bibliometric features while they used a mixture of content-based and bibliometric features. The novelty of our approach also lies in predicting citations using h-index, g-index, and e-index.

VIII. CONCLUSION

We have developed a system for predicting the number of citations of a paper at the moment of its publishing. We have shown that the accuracy of the method is mainly a function of prediction parameters and much less of used prediction algorithm. In order to avoid the huge impact of the dominant number of papers with no or a very small number of citations, we have developed an approach that targets the accurate identification of papers that will have high or highest citation impact. Our models enable both over 82.5% prediction accuracy as well as the identification of likely high impact papers at the moment of their citation before any actual citing starts.

ACKNOWLEDGMENT

We thank Saro Meguerdichian for his valuable comments.

REFERENCES