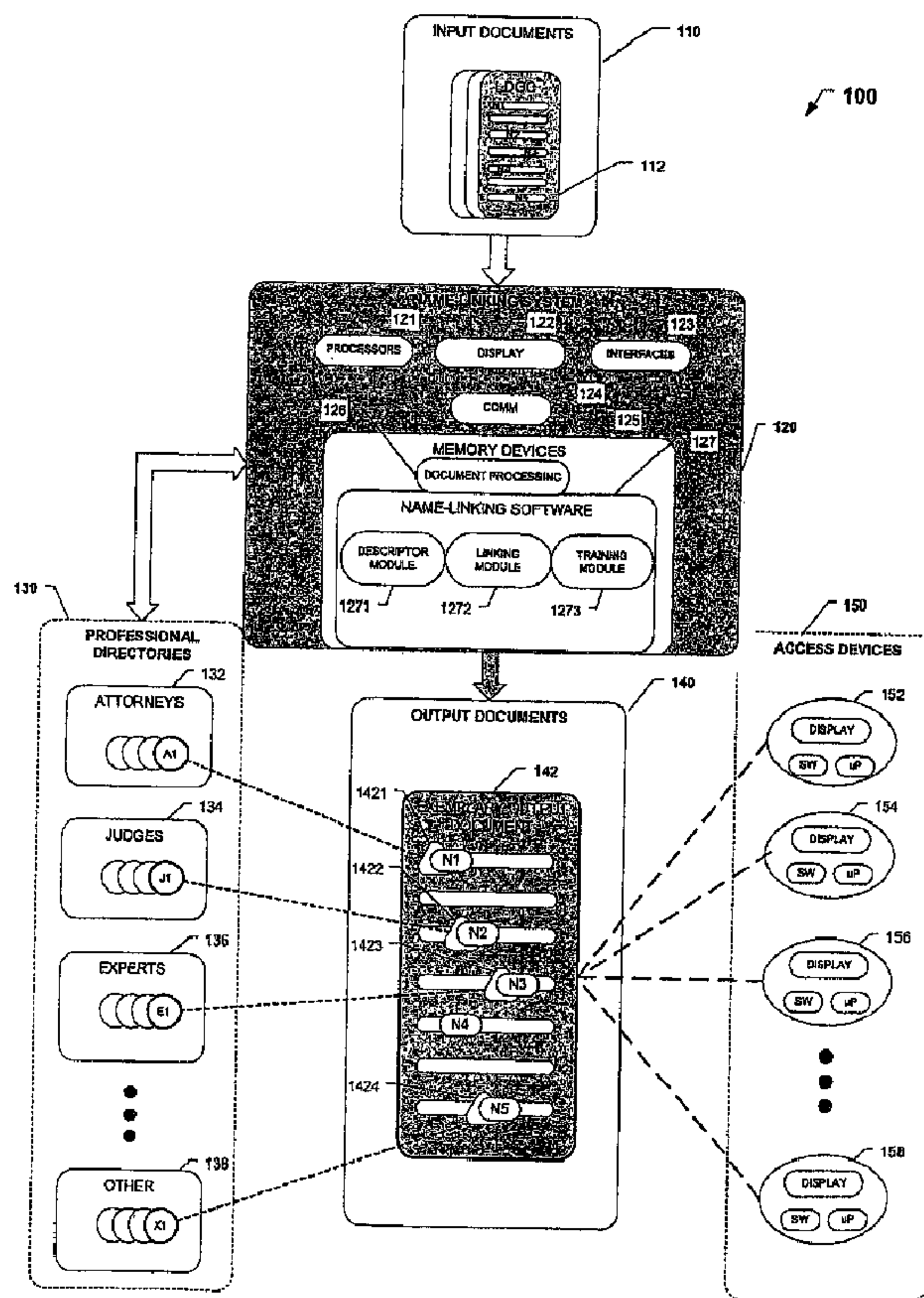




(22) Date de dépôt/Filing Date: 2002/12/20  
 (41) Mise à la disp. pub./Open to Public Insp.: 2003/07/24  
 (62) Demande originale/Original Application: 2 471 592  
 (30) Priorités/Priorities: 2001/12/21 (US60/342,956);  
 2002/06/13 (US10/171,170)

(51) Cl.Int./Int.Cl. *G06F 17/00* (2006.01),  
*G06F 17/27* (2006.01)  
 (71) Demandeur/Applicant:  
 THOMSON REUTERS GLOBAL RESOURCES, CH  
 (72) Inventeur/Inventor:  
 DOZIER, CHRISTOPHER C., US  
 (74) Agent: SIM & MCBURNEY

(54) Titre : SYSTEMES, PROCEDES ET LOGICIEL PERMETTANT D'ETABLIR DES HYPERLIENS DE PERSONNES  
 DANS DES DOCUMENTS AVEC DES ANNUAIRES PROFESSIONNELS  
 (54) Title: SYSTEMS, METHODS AND SOFTWARE FOR HYPERLINKING NAMES



(57) Abrégé/Abstract:

Hyperlinking or associating documents to other documents based on the names of people in the documents has become more desirable. Although there is an automated system for installing such hyperlinks into judicial opinions, the system is not generally

(57) **Abrégé(suite)/Abstract(continued):**

applicable to other types of names and documents, nor well suited to determine hyperlinks for names that might refer to two or more similarly named persons. Accordingly, the inventor devised systems, methods, and software that facilitate hyperlinking names in documents, regardless of type. One exemplary system includes a descriptor module and a linking module. The descriptor module develops descriptive patterns for selecting co-occurrent document information that is useful in recognizing associations between names and professional classes. The linking module tags names in an input document, extracts co-occurrent information using the descriptive patterns, and uses a Bayesian inference network that processes a (non-inverse-document-frequency) name-rarity score for each name along with the name and selected co-occurrent document information to determine appropriate hyperlinks to other documents, such as entries in professional directories.

### Abstract

Hyperlinking or associating documents to other documents based on the names of people in the documents has become more desirable. Although there is an automated system for installing such hyperlinks into judicial opinions, the system is not generally applicable to other types of names and documents, nor well suited to determine hyperlinks for names that might refer to two or more similarly named persons. Accordingly, the inventor devised systems, methods, and software that facilitate hyperlinking names in documents, regardless of type. One exemplary system includes a descriptor module and a linking module. The descriptor module develops descriptive patterns for selecting co-occurrent document information that is useful in recognizing associations between names and professional classes. The linking module tags names in an input document, extracts co-occurrent information using the descriptive patterns, and uses a Bayesian inference network that processes a (non-inverse-document-frequency) name-rarity score for each name along with the name and selected co-occurrent document information to determine appropriate hyperlinks to other documents, such as entries in professional directories.

**SYSTEMS, METHODS, AND SOFTWARE FOR  
HYPERLINKING NAMES**

**Copyright Notice and Permission**

5 A portion of this patent document contains material subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent document or the patent disclosure, as it appears in the Patent and Trademark Office patent files or records, but otherwise reserves all copyrights whatsoever. The following notice applies to this document  
Copyright © 2001, West Group.

10

**Technical Field**

The present invention concerns systems, methods, and software for hyperlinking names in documents.

15

**Background**

In recent years, the fantastic growth of the Internet and other computer networks has fueled an equally fantastic growth in the data accessible via these networks. One of the seminal modes for interacting with this data is through the use of hyperlinks within electronic documents.

20

Hyperlinks are user-selectable elements, such as highlighted text or icons, that link one portion of an electronic document to another portion of the same document or to other documents in a database or computer network. With proper computer equipment and network access, a user can select or invoke a

25

hyperlink and almost instantaneously view the other document, which can be located almost anywhere in the world. Moreover, the other document itself can include hyperlinks to yet other documents that include hyperlinks, allowing the user to "hop" around the world from document to document to document seeking  
5 relevant information at will.

More recently, there has been interest in hyperlinking documents to other documents based on the names of people in the documents. For example, to facilitate legal research, West Publishing Company of St. Paul, Minnesota provides thousands of electronic judicial opinions that hyperlink the names of  
10 attorneys and judges to their online biographical entries in the West Legal Directory, a proprietary directory of approximately 1,000,000 U.S. attorneys and 20,000 judges. These hyperlinks allow users accessing judicial opinions to quickly obtain contact and other specific information about lawyers and judges named in the opinions.

15 The hyperlinks in these judicial opinions are generated automatically, using a system that treats first, middle, and last names; law firm name, city, and state; and court information as clues to link named attorneys and judges to their corresponding entries in the professional directory. See Christopher Dozier and Robert Haschart, "Automatic Extraction and Linking of Person Names in Legal  
20 Text" (Proceedings of RIAO 2000: Content Based Multimedia Information Access. Paris, France. pp. 1305-1321. April 2000).

Although the automated system is highly effective, the present inventor recognized that it suffers from at least two limitations. First, the system exploits  
25 structural (organizational) features in judicial opinions, such as case headers, that are not common to other documents and thus limits its general application to other types of names and documents. Second, the system treats all names as equally ambiguous, or equally common, when, in fact, some names are more or less ambiguous than others. For example, the name David Smith is more  
30 common than the name Seven Drake and thus more ambiguous, or more likely to identify more than one person.

Accordingly, the present inventor has identified a need for other methods of generating hyperlinks for names, or more generally associating data that include names.

5

### Summary

To address this and other needs, the inventor devised systems, methods, and software that facilitate hyperlinking or associating names in documents, such as news articles, to or with names in other data structures, such as records in professional directories. One exemplary system includes a descriptor module and a linking module. The descriptor module develops descriptive patterns for selecting co-occurrent document information that is useful in recognizing associations between names and professional classes. The linking module tags names in an input document, extracts co-occurrent information using the descriptive patterns, classifies each names as belonging to a particular profession, and attempts to find corresponding entries in professional directories.

To find the corresponding entries, the linking module determines a name-rarity (uniqueness or ambiguity) score for each name and inputs this score along with the name and selected co-occurrent document information into a Bayesian inference network. The inference network measures the probabilities that the name refers to particular candidate records (or entries) in a given professional directory. The linking module ranks the candidate records based on the probability measurements and defines a hyperlink (or other logical association) based on the highest-ranking record that exceeds a given threshold.

The inventor also devised systems, methods, and software that facilitate searching for data that includes potentially ambiguous terms, such as names of persons or other entities. For example, one method entails receiving a query from a user, identifying one or more names in the query, assessing ambiguity or uniqueness of the names, and if the ambiguity is sufficiently great, obtaining further information and updating or supplementing the query to assist in resolving or reducing the ambiguity. The further information, which, for example, includes a professional title, location, or organization, can be obtained directly from the user or through automatic supplemental searching.

In accordance with an aspect of the present invention, there is provided a computer-implemented method comprising:

- identifying a name in a document;
- determining a rarity indicator for the name; and
- 5 defining a hyperlink for the name based on the rarity indicator.

In accordance with a further aspect of the present invention, there is provided a system for adding a hyperlink to a document including a person name, the system comprising:

- at least one processor;
- a memory coupled to the processor, the memory including instructions for:
- 10 identifying a name in a document;
- determining a rarity indicator for the name; and
- defining a hyperlink for the name based on the rarity indicator.

In accordance with a further aspect of the present invention, there is provided a data structure comprising:

- 15 a name; and
- a name-rarity indicator which indicates how likely the name is to refer to more than one entity in a population.

In accordance with a further aspect of the present invention, there is provided a computer-implemented method comprising:

- 20 determining rarity of a name in a population.

In accordance with a further aspect of the present invention, there is provided a system comprising means for determining rarity of a name in a population.

### Brief Description of Drawings

Figure 1 is a block diagram of an exemplary system 100 incorporating teachings of the present invention.

5 Figure 2 is a flow chart of an exemplary method of operating system 100 to define expressions or descriptors for use in classifying and linking names.

Figure 3 is a flow chart of an exemplary method of operating system 100 to define a hyperlink between names in a document and names in a database, based on the Bayesian inference network trained according to Figure 3 5.

10 Figure 4 is a block diagram of an exemplary Bayesian inference system used in operating system 100 to define hyperlinks.

Figure 5 is a flow chart of an exemplary method of operating system 100 to train a Bayesian inference network for use in measuring likelihood that a name in a document and a name in a database refer to the same person.

15 Figure 6 is a flow chart of an exemplary search method that incorporates teachings of the present invention.

### Detailed Description of Exemplary Embodiments

The following detailed description, which references and incorporates  
 20 Figures 1-6, describes and illustrates one or more exemplary embodiments of the invention. These embodiments, offered not to limit but only to exemplify and teach the invention, are shown and described in sufficient detail to enable those skilled in the art to make and use the invention. Thus, where appropriate to avoid obscuring the invention, the description may omit certain information  
 25 known to those of skill in the art.

### Definitions

The description includes many terms with meanings derived from their usage in the art or from their use within the context of the description. As a  
 30 further aid, the following term definitions are presented.

The terms "a" and "an" refer to at least one.



The term “or” is used in its Boolean logical sense, unless used in conjunction with “either.”

The term "document" refers to any logical collection or arrangement of machine-readable data having a filename.

5 The term "database" includes any logical collection or arrangement of machine-readable documents.

The term “hyperlink” includes any token in a document conforming structurally or functionally to any past, present, or future Uniform Resource Locator (URL) standard. It also includes any token including information  
10 identifying a specific computer system or networked device.

The term “name” includes one or more words by which an entity, such as person, animal, place, thing, group, organization or legal entity, is designated and distinguished from others.

The term “program modules” include routines, programs, objects,  
15 components, data structures, and instructions, or instruction sets, and so forth, that perform particular tasks or implement particular abstract data types. The term is not limited in terms of particular carrier medium.

#### Exemplary Computer System Embodying the Invention

20 Figure 1 shows a diagram of an exemplary computer system 100 incorporating a system, method, and software for automatically marking one or more portions of a document and defining one or more corresponding hyperlinks for each marked portion. Though the exemplary system is presented as an interconnected ensemble of separate components, some other embodiments  
25 implement their functionality using a greater or lesser number of components. Moreover, some embodiments intercouple one or more the components through wired or wireless local- or wide-area networks. Some embodiments implement one or more portions of system 100 using one or more mainframe computers or servers.) Thus, the present invention is not limited to any particular functional  
30 partition.

Generally, system 100 includes an input document database 110, a name-link subsystem 120, professional directories 130, an output documents database 140, and access devices 150.

Input documents database 110 includes one or more electronic documents, of which a document 112 is shown as representative. Document 112 includes one or more names of persons, places, things, or legal entities (more generally proper nouns, such as N1, N2, N3, N4, and N5 distributed throughout the document. In the exemplary embodiment, document 112 is an electronic version of a written news article or other text document, for example, a judicial opinion or other type legal document. However, in other embodiments, document 112 includes one or more images or multimedia data containing one or more names.

Coupled to database 110 is a computerized name-linking system 120. System 120 includes one or more conventional processors 121, display device 122, interface devices 123, network-communications devices 124, memory devices 125, document-processing software 126 and marking-and-linking software 127. Software 126 and 127 includes various software and data components which can take a variety of forms, such as coded instructions or data on an electrical, magnetic, and/or optical carrier medium, and which can be installed on system 120 separately or in combination through a network-download or through other software transfer methods.

Examples of document-processing software include word-processing programs, HTML-editing programs, spreadsheet programs, electronic-mail programs, presentation-development programs, browsing programs, document-management programs, and file-backup programs. Thus, the invention is not limited to any particular genus or species of document-processing software.

In the exemplary embodiment, software 127 is an add-on tool to document-processing software 126. However, in other embodiments, it operates as a separate application program, such as a network-accessible program, or as part of the kernel or shell of an operating system. More particularly, software 127 includes a descriptor module 1271, a linking module 1272, and a training module 1273, all of which are described in further detail below.

System 120 is coupled to professional directories 130 and output documents database 140.

Professional directories 130 include one or more professional directories, such as an attorney directory 132, a judge directory 134, an expert database 136, and an other-professionals directory 138. Each directory (or more generally database) includes a set of records or other data structures containing information associated with one or more named or identified entities, such as persons, places, things, or legal entities. For example, attorney directory 132 includes a number of attorney records, such as exemplary attorney record A1; judge directory 134 includes a number of judge records, such as exemplary judge record J1; expert directory 136 includes a number of expert records, such as exemplary expert record E1; and other directory includes a number of records which contain information associated with other individuals, such as doctors, professors, accountants, teachers, celebrities, etc. Some embodiments may include databases of telephone and e-mail addresses, credit reports, tax reports, criminal records, medical information, school records, etc.

Output documents database 140 includes one or more processed documents, such as an exemplary document 142. Document 142 includes marked names N1, N2, N3, N4, N5 and respective hyperlinks 1421, 1422, 1423, 1424, and 1425, which each reference a biographical record or other data structure within at least one of professional directories 130, or to one of the professional directories without designation of a particular directory record, or to a subset of records in a directory. Hyperlinks 1421-1425, generated by name-link processor 120 and embedded within or otherwise associated with the document, are selectable to link respective marked name portions N1, N2, N3, N4, N5 of document 140 to databases 130, 132, and 134 via a public or private local- or wide-area network or dedicated communications path (not shown.) The exemplary embodiment presents the marked names in a contrasting color or font, or in other user-detectable ways, to signify its association with an existing hyperlink. Documents within output database 140 are accessible over a local- or wide-area network through access devices 150.

The exemplary embodiments provides output database 140 as part of a web server, such as a Microsoft Internet Information Server 4.0 running on a network of several servers with processors and extended memory and disk configurations. Database 140 can take on any number of forms on a variety of computer platforms. Moreover, in some embodiments, database 140 includes  
5 redundant content to allow for more than one device, like access devices 150, to simultaneously access multiple copies of the same document.

Access devices 150 include exemplary access devices 152, 154, 156, and 158. Each access devices includes a display, a processor (uP), and software  
10 (SW). The term "access device," as used herein, encompasses browser-equipped personal computers, network appliances, personal digital assistants, telephones, cell phones, web phones, televisions, web television, etc. It also includes monitors and other types of equipment which can output data in a form with which users or other computers can interact. Thus, the present invention is not  
15 limited to any particular class or form of access device.

#### Exemplary Operation of System 100

In general, exemplary operation of system 100 entails operation of descriptor module 1271, linking module 1272, and training module 1273.  
20 Descriptor module 1271 generates one or more professional-name descriptor structures or patterns for use in identifying names that are likely to refer to individuals within one or more particular professional classes (or satisfy other predetermined criteria.) Linking module 1272 receives an input document, such as document 110, and hyperlinks one or more names in the input document to  
25 one or more professional directories, based on professional-name descriptor structures and/or other data extracted from document 110 and input into a Bayesian inference network. Training module 1273 defines the conditional probabilities at various nodes in the Bayesian inference network used by linking module 1272.

30

### A. Descriptor Module Structure and Operation

More particularly, Figure 2 shows a flowchart 200 illustrating an exemplary method of operating descriptor module 1271 to generate name descriptors for a given profession. One premise of the exemplary embodiment is that some personal names are much more likely to pertain to a single individual than other names and that when such names are also associated with a common profession (or other classification), one can automatically identify descriptive language common to people in the profession (or classification). This language then could be used to identify most members of the profession (or classification) who are named in the corpus.

Flow chart 200 includes process blocks 210-260. Though these blocks (and those of other flow charts in this document) are arranged serially in the exemplary embodiment, other embodiments may reorder the blocks, omit one or more blocks, and/or execute two or more blocks in parallel using multiple processors or a single processor organized as two or more virtual machines or subprocessors. Moreover, still other embodiments implement the blocks as one or more specific interconnected hardware or integrated-circuit modules with related control and data signals communicated between and through the modules. Thus, this and other exemplary process flows in this document are applicable to software, firmware, hardware, and other types of implementations.

Block 210 entails identifying names in a collection of documents (or corpora) that match names in a professional directory and names that do not match. Identifying matching and non-matching names --that is, in-directory and out-of-directory names, entails identifying all names in the collection using a name-tagging program and then executing a search against a professional directory, such as one of professional directories 130. Although the present invention is not limited to any genus or species of name tagger, suitable examples of name taggers include the NetOwl parsing software from IsoQuest, Inc. of Fairfax, Virginia. (The inventor contemplates a name tagger based on a maximum-entropy model for some embodiments.)

After executing the search, the exemplary embodiment identifies a subset of the in-directory names as rare or unique in-directory names. This entails

computing name- uniqueness probability for each in-directory name, with the name-uniqueness probability based on a language model for the names in the directory. The exemplary language model is defined in terms of first-name probability and last-name probability, with each first-name and last-name probability based respectively on the ratio of the total number of occurrences of the first name and last name to the total number of names in a list of names drawn from the general population.

The list of names should be large enough to represent accurately the distribution of names in the general population. If the professional directory or other database is large enough, it can be used as the basis for the language model. If the professional directory is small, the language model should be based on some other list such as the list of licensed professionals listed in public records across the United States. The description that follows assumes that the professional directory (or database) in question is large enough to be representative of the names in the general population.

Once the language model is defined, the name-match probability score for each in-directory name is computed using

$$P(\textit{name}) = P(\textit{first name}) \cdot P(\textit{last name}) \quad (1)$$

where  $P(\textit{first name})$  denotes the probability of drawing the first name at random from all the first names in the directory and  $P(\textit{last name})$  similarly denotes the probability of drawing the last name at random from all the last names in the directory. A probability of name uniqueness or rarity is then computed as

$$P(\textit{nameUniqueness}) = \frac{1}{(H \cdot P(\textit{name})) + 1} \quad (2)$$

where  $H$  denotes the size of the human population likely to be referenced in the corpus. For example, for a corpus consisting of Wall Street Journal articles,  $H$  is taken to be 300 million, the approximate population of the United States. Each in-directory name that has a name-uniqueness probability exceeding a threshold value, such as  $0.07$ , is then used as a basis, along with the out-of-directory names for further processing at block 220.

Block 220 extracts document information co-occurring or otherwise associated with one or more of the identified in-directory names and one of more

of the out-of-directory names. In the exemplary embodiment, this entails extracting text or information within a certain textual window (or document region) around each of the rare in-directory names and around all the out-of-directory names.

5 More specifically, exemplary extraction entails extracting unigrams and bigrams occurring within a textual window extending eight words before and eight words after each occurrence of identified rare in-directory and out-of-directory names in the corpus. (Other embodiments use other sizes and forms of text windows, such as documents grammatical or organizational structures. For  
10 instance, some embodiments define the window based on number of characters, sentence, or subsection.) The unigrams and bigrams associated with in-directory names are defined as co-occurring in-directory unigrams and bigrams, whereas those associated with out-of-directory names are referred to as co-occurring out-of-directory unigrams and bigrams. Execution continues at block 230.

15 Block 230 determines a probability or likelihood that the extracted information will occur with an in-directory name rather than an out-of-directory name. In the exemplary embodiment, this entails computing the probability of each in-directory unigram and bigram occurring within an eight-word window before and after the out-of-directory names, and the probability of each out-of-  
20 directory unigram and bigram occurring within an eight-word window before and after the in-directory names. These co-occurrence probabilities are computed as

$$P(\text{unigram} / \text{in} - \text{directory rare name}) = \frac{IU}{NI} \quad (3)$$

where IU = number of times unigram occurs in window with an in-directory rare  
25 name and NI = number of in-directory rare names.

$$P(\text{unigram} / \text{out} - \text{of} - \text{directory rare name}) = \frac{OU}{NO} \quad (4)$$

where OU denotes the number of times unigram occurs in window with an out-of-directory name and NO denotes the number of out-of-directory names.

$$P(\text{bigram} / \text{in} - \text{directory rare name}) = \frac{IB}{NI} \quad (5)$$

where IB denotes the number of times bigram occurs in window with in-directory rare name and NI the number of in-directory rare names.

$$P(\text{unigram} / \text{out - of - directory rare name}) = \frac{OB}{NO} \quad (6)$$

where OB denotes the number of times bigram occurs in window with an out-of-directory name and NO denotes the number of out-of-directory names.

To determine the likelihood of unigrams and bigrams co-occurring with an in-directory rare name rather than an out-of-directory name, the exemplary embodiment divides the in-directory rare name co-occurrence probability by the respective the out-of-directory co-occurrence probability. These likelihood formulations are expressed as

$$L(\text{unigram}) = \frac{P(\text{unigram} / \text{in - directory rare name})}{P(\text{unigram} / \text{out - of - directory rare name})} \quad (7)$$

$$L(\text{bigram}) = \frac{P(\text{bigram} / \text{in - directory rare name})}{P(\text{bigram} / \text{out - of - directory rare name})} \quad (8)$$

where  $L(\text{unigram})$  is the likelihood of co-occurrence of a given unigram with an in-directory rare name and  $L(\text{bigram})$  is the likelihood of co-occurrence of a given bigram with an in-directory rare name.

Block 240 ranks the extracted information based on the co-occurrence likelihoods of preceding and trailing unigrams and bigrams. To this end, the exemplary embodiment ranks or sorts the co-occurring indirectory unigrams and bigrams in descending order of their in-directory rare name co-occurrence likelihoods. (Other embodiments may compute and use co-occurrence probabilities.) Examples of ranked lists of preceding and trailing unigrams and bigrams for legal professionals along with their likelihood scores are shown in the following two tables.

Preceding Unigram	Likelihood Score	Trailing Unigram	Likelihood Score
Solicitor	74.32	Judges	46.45
Judge	44.69	Attorney	22.40
Lawyer	18.94	Lawyer	18.34
Counsel	10.22	Prosecutor	12.38
congressman	10.13	Attorneys	9.29



Attorney	6.89	Counsel	8.37
Prosecutor	6.75	Judge	7.96

Table 1: Preceding and Trailing Unigram Examples

Preceding Bigram	Likelihood Score	Trailing Bigram	Likelihood Score
District judge	152.37	lead counsel	92.91
Court judge	106.85	Tax lawyer	55.75
General counsel	15.03	senior attorney	55.75
city attorney	14.86	Former congressman	55.75
democratic leader	14.45	u.s. attorney	37.16
District attorney	12.38	Law professor	18.58

Table 2: Preceding and Trailing Bigram Examples

5           Block 250 entails selecting one or more set of the extracted information based on the rankings. In the exemplary embodiment, this is a manual selection process; however, other embodiments may apply an automatic selection criteria based, for example, on a specific minimum rank or a specific minimum rank in combination with a minimum threshold.

10           More specifically, the exemplary embodiment selects two sets of terms, referred to as anchor terms. The first set of anchor terms include terms that have a high likelihood of preceding an in-directory rare name, and the second set of anchor terms include terms that have a high likelihood of trailing such a name. In many instances, the anchor terms are the nouns in appositive phrases or  
15 clauses that precede or trail a given name. The exemplary embodiment also groups lowercase terms that are synonymous with each other. For example, terms such as, "lawyer," "counsel," and "prosecutor" are considered synonymous with "attorney," and are thus grouped together to form a single anchor term to reduce the number of regular expressions or descriptors generated at block 260.

Block 260 entails automatically generating profession descriptor patterns (or expressions) that correlate with membership in the profession. The exemplary descriptors represent grammatical patterns in corpus fragments that are bounded by in-directory rare names and the selected anchor terms.

5           The exemplary embodiment generates the patterns in the following way. For each sentence fragment, bounded by an anchor term and a rare name, the exemplary embodiment derives a generalized pattern (or structure) by splitting the fragment on space characters and commas into separate tokens and then normalizing each token based on its most commonly associated part of speech in  
10 an English lexicon. Tokens outside the lexicon are typed "other," and forms of the verb "to be" are typed "is" to distinguish them from other verbs. Punctuation marks other than commas are typed as "other." The exemplary embodiment uses a publicly available English lexicon of about 90,000 words, with each word associated with one or more parts of speech, listed in order of frequency within a  
15 corpus. An exemplary vocabulary for the descriptive patterns is shown in Table 3.

Pattern Symbol	Grammar Feature	Examples
\$det-	determiner	a, an, the
\$pos	Possessive	his, our
\$adj	Adjective	legal
\$noun	Noun	attorney
\$prep	Preposition	for
\$pronoun	Pronoun	he
\$properNoun	proper noun	Johnsons
\$adv	Adverb	legally
\$inf	Infinitive marker	to
\$isVerb	forms of verb "to be"	is, was
\$verb	Verb	said
\$comma	Comma	,
\$other	All other parts of speech or punctuation	

/s	space	
----	-------	--

Table 3: Exemplary Vocabulary for Descriptive Patterns

Next, the exemplary embodiment creates a merged list of unique patterns and counts how often each is repeated in the corpus. The patterns are then sorted based on their occurrence counts, and singular patterns are discarded. The remaining patterns are then taken as indicative of membership in the given profession.

Inset box 260' shows that an exemplary pattern structure includes a name structure 261, grammar structures 262, an anchor-terms structure 263, and relative positional data 264 and 265. Name structure 261 is a placeholder for a name in a fragment. Grammar structures 262 include punctuation, parts-of-speech identifiers, and associated positional information indicative of any existing grammatical structural patterns between the name and anchor-terms structure 263. Anchor-terms structure 263 represents and includes one or more anchor terms.

Relative positional data 264 and 265 respectively indicate the relative position of name structure 261 and grammar structures 262, and grammar structure 262 and anchor terms 263. Although the exemplary embodiment implements the relative positional data implicitly in terms of the ordering of the data within the pattern structure, other embodiments explicitly denote the relative position as "before" or "after." Some embodiments omit the name structure and/or intervening grammar structures and define patterns in terms of word or character distance between anchor terms and an implicit name structure.

Examples of profession-descriptive patterns for identifying attorneys from a corpus of news articles are shown in the following table. Also shown are corresponding sentence fragments extracted from a document using these patterns.

Attorney Descriptor Patterns	Example Retrieved Fragments
\$Anchor \$comma \s+ \$Name	attorney, <name >
\$Anchor \s+ \$Name	Attorney general <name>

\$Anchor \s+ \$prep \s+ \$det \s+ \$other \$comma \s+ \$Name	attorney for the Johnsons, <name>
\$Name \$comma \s+ \$det \$Anchor	<name>, a lawyer
\$Name \$comma \s+ \$pos \s+ \$Anchor	<name>, his attorney
\$Name \$comma \s+ \$det \s+ \$adj \s+ \$Anchor	<name>, a defense attorney

Table 4: Examples of Attorney-Descriptive Patterns and Sentence Fragments

In essence, the exemplary embodiment uses rare or less ambiguous names as virtual tags to identify or filter out a set of exemplary sentence fragments that contain descriptions of named professionals and that can be used as the basis for defining grammatically descriptive patterns. The rarity filter serves to identify good examples much more efficiently than looking at sentence fragments around personal names in general. These patterns are then used to assist in automatic generation of hyperlinks in linking module 1272.

10

#### B. Linking Module Structure and Operation

Generally, linking module 1272 (in Figure 1) receives an input document, such as document 110, and hyperlinks one or more names in the input document to one or more professional directories, based on the professional-name descriptor structures defined by descriptor module 1271.

15

Figure 3 shows a flowchart 300 of an exemplary method embodied in linking module 1272. Flowchart 300 includes process blocks 310-370.

Execution of the exemplary method ensues at block 310 which entails receiving a document, such as document 112 from input documents database 110. In some embodiments, linking module 1272 executes in the context of one or more sessions initiated by document-processing software 127, and the input document represents an entire document hosted by the document-processing software or one or more selected portions of the document within an active processing window in a document-processing program. However, in other embodiments, the document or document portion is received or retrieved from a currently active processing window in a document-processing program. In some

20

25

other embodiments, the linking module is a stand-alone application which interacts with one or more databases coupled to a host computer system, such as workstation. Execution continues at block 320.

In block 320, the linking module uses a generic tagger to tag or mark  
5 each person, place, and organization name in the input document. The exemplary embodiment uses the same tagger as used in the descriptor module. In the exemplary embodiment, tagging the names also entails resolving apparent co-references to the same name within the input document.

To this end, the exemplary embodiment links references within a  
10 document to the same name together, using a set of name comparison rules. These links are termed in-document co-reference chains, or name chains. During the tagging process, the last name of each newly found name is compared against the last names of those names already found. If no matches are found, the newly found name is treated as unique name in the document. If  
15 the newly found last name matches an existing last name, the first name paired with the newly found last name is compared to the first name paired with the matching last names. If a first name matches or is compatible, the names are assumed to be the same. If the first name is incompatible with the first name of any of the matching last names, the name is treated as unique in the document.  
20 If the newly found name has no clearly associated first name, then the name is treated as a reference to the most recently found name having the same last name. Other embodiments may use other techniques to resolve in-document co-references.

In block 330, the linking module determines which of the one or more  
25 tagged name chains is likely to be associated with one or more of the professional directories. In the exemplary embodiment, this entails applying one or more of the profession-descriptive patterns generated by descriptor module 1271. In applying the descriptive patterns, the exemplary embodiment tries to match or map each descriptive pattern for a given profession to the text around  
30 each occurrence of a tagged name in the input document. If the descriptive pattern successfully maps to an occurrence of a name, the name (and its in-document co-references) is deemed a candidate professional and added to a list

of candidate professionals of a given type. (Other embodiments exploit the regular structure or cue features of documents as further evidence regarding the nature of names in the documents. For examples, judicial opinions (caselaw), include case headers, counsel phrases, concurring opinions, and dates that can be  
 5 used to identify and/or distinguish attorneys and judges.) The list of candidate name chains is forwarded for further processing at block 340.

Block 340 entails defining one or more name templates or other data structures based on the tagged name chains of candidate professionals and related co-occurring text or other information from the document. In the  
 10 exemplary embodiment, defining the name templates entails forming for each tagged name chain a collection of sentences in the document that contain a name in the chain, and a collection of paragraphs in the document that contain a name in the chain. In some embodiments, the collections of sentences and paragraphs are document position indicators that denote the start and/or end of sentences  
 15 and paragraphs.

After forming the sentence and paragraph collections, the linking module defines a template for each name chain, with each template having the form of exemplary name template 340'. Name template 340' includes a name record 341, a descriptor data record 342, an location data record 343, an organization  
 20 record 344, and a name uniqueness (or rarity) record 345.

Name record 341 includes a first-name field (first), a middle-name field (mid), a last-name field (last), and a suffix field (suffix). Other embodiments include other name-related data such nicknames from a lookup table or common alternative spellings. And still other embodiments exclude one or more of the  
 25 fields used in the exemplary embodiment, such as suffix.

Descriptor data record 342 includes one or more descriptor fields, such as D1, D2, D3... Dn. Each descriptor field, in the exemplary embodiment, includes sentence fragments extracted from the input document using one or more of the profession-descriptive patterns generated by descriptor module 1271 for a given  
 30 profession. For a document where a name is repeated or co-referenced at multiple places, the descriptor structures are applied to each occurrence of the name to assemble the set of descriptor fields. Some embodiments may even

apply the descriptor patterns to pronoun references to a given name. (In other embodiments, the descriptors also include verbs that co-occur within a certain distance of the names.)

Application of the patterns entails aligning the name portion of each  
 5 pattern with the names in the sentence collection and determining if the grammar of the adjacent portion (fragment) of the sentence agrees with the grammar of the pattern. If there is agreement, the exemplary linking module copies one or more portions of the corresponding sentence fragment, such as the anchor term, into a respective descriptor field of the descriptor data record 342.

10 Location data record 343 includes a same-sentence field and a same-paragraph field. The same sentence field includes subfields LS1, LS2, ..., LSj, with each LS subfield including a location name that co-occurred in a sentence with a reference to the name in name data record 341. The same-paragraph field includes subfields LP1, LP2, ..., LPk. Each LP subfield includes a location  
 15 name that co-occurred in a paragraph containing a reference to the name in data record 341. In the exemplary embodiment, constructing this record entails searching for locations in the sentence and paragraph collections for a given name and copying found locations into the respective subfields.

Organization data record 344 includes a same-sentence field and a same-  
 20 paragraph field. The same sentence field includes subfields OS1, OS2, ..., OSj, with each subfield including an organization that co-occurred in a sentence with a reference to the name in name data record 341. The same-paragraph field includes subfields OP1, OP2, ..., OPk. Each OP subfield includes an organization that co-occurred in a paragraph containing a reference to the name  
 25 in data record 341.

In the exemplary embodiment, a sentence is understood to include grammatical as well as typographical sentences, and a paragraph includes any group of one or more sentences that are delimited or separated from another group of one or more sentences by punctuation or other signaling device or  
 30 technique. Also, in the exemplary embodiment, the same-sentence and same-paragraph locations and organizations need not be mutually exclusive. That is, if a term occurs in a sentence with an occurrence of a name, it also occurs within

same paragraph containing the sentence. However, in other embodiments, the same-sentence and same-paragraph locations could be defined as mutually exclusive.

Although the exemplary embodiment provide implicit positional  
 5 information for the locations and organizations through the structure of the same-field and same-paragraph fields, other embodiments may use other techniques to incorporate position information relative to a given name in the name template. For example, some embodiments use character, word, sentence, paragraph, or page offset from the name, or more precisely a given name  
 10 occurrence. Other embodiments provides positional information as a set of binary codes or flags, with each flag indicating whether a logically associated or corresponding location or organization occurred within a sentence or paragraph with its associated name. Still other embodiments provide flags that indicate whether the places are within or without a certain text window or document  
 15 region relative to a name. Yet, other embodiments provide relative document position or absolute document position for each organization or place.

Name uniqueness (or rarity) record 345 includes a score field which contains an indicator of uniqueness or rarity of its associated name. In the exemplary embodiment, this indicator is a numerical quantity indicative of a  
 20 prior probability of a name matching a candidate record drawn from a particular directory. More specifically, the exemplary embodiment defines the quantity as the probability of name uniqueness and computes it using

$$P(\textit{nameUniqueness}) = \frac{1}{(H' \cdot P(\textit{name})) + 1} \quad (9)$$

where  $H'$  denotes the size of the professional class indicated by the descriptor  
 25 match and  $P(\textit{name})$  is defined as

$$P(\textit{name}) = P(\textit{first name}) \cdot P(\textit{last name}) \quad (10)$$

where  $P(\textit{first name})$  denotes the probability of drawing the first name at random from all the first names in a name list representative of the general population and  $P(\textit{last name})$  similarly denotes the probability of drawing the last name at  
 30 random from all the last names in a name list representative of the general population.



Although the exemplary embodiment uses a template, such as name template 340' for multiple types of professions, some embodiments may omit or add other template features. For example, templates for judges may omit separate location information, since organization information, such as a court name, implicitly contains location information. Other embodiments may omit  
5 information altogether rather than just its explicit form.

Block 350, which is executed after defining the name templates in block 340, retrieves a set of candidate directory entries from one or more of professional directories 130. To this end, the exemplary embodiment searches  
10 for directory entries that have the same last name as one of the candidate professional names. It then retrieves full name, title, organization, location, and entry identification information for these candidate directory entries for further processing in block 360.

Block 360 entails comparing and scoring the similarity of each name  
15 template to one or more of the retrieved candidate records or data structures. In the exemplary embodiment, this entails use of one or more Bayesian inference systems, such as the one shown in Figure 4.

Figure 4 show an exemplary inference system 400 which includes input name template 410, and an input candidate record 420, and one or more  
20 Bayesian inference engines, such as Bayesian inference engine 430.

Input name template 410 includes name data 411, descriptor data 412, location data 413, organization data 414, and name uniqueness data 415. Input candidate record 420 includes name data 421, title data 422, location data 423, organization data 424, and record identification data 425.

Inference engine 430 includes evidence comparison modules 431-434  
25 and calculation module 435. Comparison modules 431-434 include respective data and logical rule structures which define various comparison states and associated probabilities. In the exemplary embodiment, each inference engine is tailored to a specific professional directory or other database. Also, the  
30 exemplary embodiment implements each engine using a re-configurable software module with configuration options for defining the comparison logic

and calculations. However, other embodiments may use entirely distinct inference-engine structures.

Each comparison module generally includes two or more mutually exclusive states indicating a potential result of comparing a candidate data item to a respective input data item. Each state is associated with specific comparison logic and conditional probabilities for the state given that a candidate record matches the input name template and given that the candidate records do not match the input name template. (Some embodiments include multiple sets of states, logic, and conditional probabilities, with each set associated with a particular professional directory or profession.) Exemplary states include: an exact match, a strong-fuzzy match, a weak-fuzzy match, an unknown (or unspecified) match, and a mismatch.

An exact match occurs when the data items or elements match exactly. A strong-fuzzy match occurs when the elements fall short of matching exactly, but are strongly compatible. A weak fuzzy match occurs when the elements fall short of matching exactly, but are weakly compatible. An unknown match occurs when there is insufficient information to determine whether the data match or mismatch. And, a mismatch occurs when the items have no compatibility.

More precisely, the exemplary embodiment defines each of the states for each evidence comparison module as follows: Comparison module 431 has three states: an exact-match (EX) state; a strong-fuzzy (SF) state; and a weak-fuzzy (WF) state. For an exact match to occur all the components of an extracted name must exactly those of a candidate record. For example, Abraham Lincoln exactly matches Abraham Lincoln, but not Abe Lincoln, Abraham Lincoln, Jr., or Abraham S. Lincoln. For a strong-fuzzy match, the first and last name in the tagged document must match the first and last name in the record, with all other components in the record being unspecified or blank. Thus, Abraham Lincoln is a strong fuzzy match with Abraham Lincoln, Jr. and Abraham S. Lincoln. For a weak fuzzy match, only the last name matches while all other components are unspecified or have matching variant forms such as nicknames. Thus, Abraham Lincoln is a weak fuzzy match to Abe Lincoln.

(Although not shown in the Figures, comparison 431 exemplarily includes or has access to a database of names and common nicknames or variants, which it accesses when name comparison reveals that first names do not match exactly. Other embodiments, however, may include such nicknames into the name template itself or even omit nickname consideration completely.)

Comparison module 432 includes four states: an exact-match (EX) state, a strong-fuzzy (SF) state, a weak-fuzzy (WF) state, and a no match (NO) state. An exact match occurs where all elements of a fully specified descriptor match. For example, the descriptor "U.S. Supreme Court Justice" exactly matches the title "U.S. Supreme Court Justice." A strong fuzzy match occurs where some but not all of the elements of a descriptor and title match. For example, the descriptor "Supreme Court Justice" is a strong fuzzy match to the title U.S. Supreme Court Justice. An unknown state occurs when the descriptor identifies a general profession that is consistent with the profession(s) covered by the directory. For example, if the descriptor is "judge" and the title in the directory entry is "U.S. 8th Circuit Court judge," then the match is deemed unknown or unspecified. A mismatch or "no match" state occurs when the descriptor conflicts with or inconsistent with the specification in the record. For example, if the descriptor is New York District judge and the title in the directory entry is "U.S. 8th Circuit Court judge," then the match state is mismatch.

Comparison module 433, which compares location evidence, has five match states: an exact-match (EX) state, a strong-fuzzy (SF) state, a weak-fuzzy (WF) state, an unknown or unspecified (UN) state, and a no match (NO) state. An exact match occurs where city and state location evidence that is explicitly linked within the document to the extracted name matches the city and state of a directory record. An explicit linkage occurs, for example, when the location co-occurs in the same sentence. A strong-fuzzy match occurs when the city or state that occurs in the same paragraph with the extracted name matches the corresponding city or state in a candidate directory. A weak-fuzzy match occurs when the city or state that occurs in the same document but outside the same paragraph of the extracted name matches the city or state listed in a directory entry. The unknown, or unspecified, state occurs when the extracted name is not

explicitly linked to a particular city or state and none of the locations in the text match the city or state information in the candidate record. A mismatch occurs when the extracted name is explicitly linked to a city or state name that does not match the city or state information in the candidate record.

5           Comparison module 434, which compares organization data, has five states: an exact-match (EX) state, a strong-fuzzy (SF) state, a weak-fuzzy (WF) state, an unknown or unspecified (UN) state, and a no match (NO) state. An exact match occurs when an extracted name is explicitly linked to a particular organization in the text and that organization matches the organization in the  
10 candidate record. An explicit linkage occurs, for example, when the location co-occurs in the same sentence. For example, an exact match exists if text describes an attorney as working at the firm of Smith & Jones and the directory entry lists Smith & Jones as an associated law firm or other organization. A strong-fuzzy match occurs if the extracted name occurs in the same paragraph as the extracted  
15 organization and if the extracted organization matches the organization in a candidate record. A weak-fuzzy match occurs if the extracted name occurs in the same document but outside the paragraph and if the extracted organization matches the candidate organization. An unknown match occurs when the extracted name is not explicitly linked to a particular organization and none of  
20 the organization names in the text matches the organization information in the candidate record. And, a mismatch occurs when the extracted name is explicitly linked to an organization name and that organization name does not match the organization name in the candidate record.

          The table below summarizes the various states that are in effect for each  
25 of the evidence comparison modules in the exemplary Bayesian inference engine.

	States⇒ Evidence ⇓	Exact Match	Strong Fuzzy Match	Weak Fuzzy Match	Unknown	No Match
E1	Name	✓	✓	✓	--	--
E2	Descriptor	✓	✓	--	✓	✓
E3	Organization	✓	✓	✓	✓	✓
E4	Location	✓	✓	✓	✓	✓

Evidence comparison modules 431-434 conduct their respective comparisons and output their results in the form of eight conditional probabilities to calculation module 435.

Calculation module 435 calculates a similarity score or match probability based on these conditional probabilities and the uniqueness or name-rarity data for the input name template. The exemplary calculation uses the following form of Baye's rule:

$$P(M|E) = \frac{P(M) \prod_{i=1}^4 P(E_i|M)}{P(M) \prod_{i=1}^4 P(E_i|M) + P(\neg M) \prod_{i=1}^4 P(E_i|\neg M)} \quad (11)$$

where  $P(M|E)$  denotes the probability that a template matches a candidate record given a certain set of evidence, such as an input name template and a candidate record.  $P(M)$  denotes the prior probability that a template and biography record match (that is, refer to the same person), and  $P(\neg M)$  denotes the prior probability that a template and biographical record do not match. The exemplary embodiment defines  $P(M)$  as the probability of name uniqueness or name-rarity within the professional population, and  $P(\neg M)$  as

$$P(\neg M) = 1 - P(M) \quad (12)$$

$P(E_i|M)$  is the conditional probability that  $E_i$  takes on a particular state given that the input name template matches the candidate record. For example, if  $E_3$  denotes location match evidence, then  $P(E_3|M)$  denotes the probability that the

location information in the name template and candidate record have the match state determined by comparison module 433 (exact match, strong fuzzy mach, weak fuzzy match, unknown match, or no match) given that a name template and candidate record match.  $P(E_i|\neg M)$  denotes the conditional probability that  $E_i$  takes on a particular state given that a name template does not match any record in the professional directory. For example,  $P(E_3|\neg M)$  denotes the probability that the location information in a person template and candidate record match given that the template and candidate do not match. (Other embodiments include lesser or greater numbers of states, as well as other types of co-occurring information.)

The score for each candidate record-name-template comparison is associated with a candidate record identification 424 for the candidate record. In Figure 4, this association represented by the broken line between score 440 and candidate record identification 425. In Figure 3, execution continues at block 370.

In block 370, the linking module links one or more of the professional names tagged in the input document to one or more of the candidate professional directories based on the comparison scores. In the exemplary embodiment, this entails executing process blocks 371-375 shown in inset 370'.

Block 371 selects the best candidate record from among the candidate records for a particular name template. Specifically, this entails selecting the candidate having the highest comparison score. If there is no single candidate having the highest score, the exemplary embodiment advances to block 372 without selection of a candidate for the name template. However, other embodiments could employ some form of tie breaker (for example one based on chronology), or construct links to each of the highest ranking candidate records with a message qualifying uncertainty in the link accuracy, or build a link that presents a menu of the highest ranking candidates for the user.

Block 372 determines if the selected candidate record meets other criteria. To this end, the exemplary embodiment determines if the score of the selected candidate record satisfies a given threshold criterion, such as greater than or equal to 0.05. However, other embodiments use additional criteria, such

as commercial or chronological relationship with the professional directory containing the candidate record. If the candidate record satisfies the link criteria execution, continue at block 373.

Block 373 entails building a hyperlink that links at least one occurrence  
 5 of the name under consideration in the input document to the entry in the professional directory hosting the candidate record. In the exemplary embodiment, this includes marking all occurrences of the name in the document and embedding a URL (uniform resource locator) identifying the candidate record in the document. (In some embodiments, the candidate reference  
 10 identification and a document identification number for the input document are written into an index that can be used to facilitate name-based searching and further document linking between the professional directories and document databases.) The marking can take any number of forms, such as a font change relative to other text in the document. Other embodiments may insert a  
 15 hyperlink that references the professional directory, instead of a particular record in the directory. Other templates include character-offset information for use in placing hyperlinks at the correct spot within a given sentence, paragraph, or document. Still other embodiments may define the hyperlink in terms of one or more intermediate destinations that are forwarded or otherwise routed to the  
 20 directory or directory record to effect the desired linkage.

After building the link in block 373 (or after determining that the link criteria is not met at block 372), execution proceeds to block 374. Block 374 determines whether the input document contains another name for possible linkage. An affirmative determination returns execution to block 371 for  
 25 selection of a candidate record for another name template, and a negative determination branches execution to block 375. Block 375 returns execution to block 310 to receive another input document for further processing.

### C. Training Module Structure and Operation

30 Figure 5 shows a flowchart 500, which illustrates exemplary structure and operation of training module 1273 to define the conditional probabilities

used in the Bayesian inference engine of Figure 4. Flowchart 500 includes process blocks 510-560.

At block 510, execution begins by receiving a set of training documents having tagged names that are known to match names in a professional directory. In the exemplary embodiment, the training documents are manually tagged; however, in other embodiments the documents may be automatically tagged. The training documents may be held in a local or remote database and communicated to the training module via a variety of transfer techniques.

Block 520 entails extracting data from the training documents based on the tagged names. To this end, the exemplary embodiment generates a name template, using the extraction module from linking module 1272, for each tagged name, with each template including an extracted name, extracted descriptor text, extracted location list, an extracted organization list. The name template has a structure similar to that of name template 340' in Figure 3.

Block 530 entails searching one or more professional directories, based on the names in the training data structures. In the exemplary embodiment, this entails searching one or more of the professional directories and retrieving directory entries with last names that match the last names in the name templates for further processing.

Block 540 entails determining conditional probabilities for each state of each evidence variable. In the exemplary embodiment, this determination entails determining frequency counts for each comparison state based on the name templates for the manually tagged set of documents. Specifically, for cases in which a tagged name has been manually matched to a candidate record, the embodiment counts the number of times each particular evidence state occurs for each of the evidence variables: name, descriptor, location, and organization. The embodiment then divides the count for each evidence state by the total number of matches to obtain  $P(E_i|M)$  that is, the conditional probability values for each state given a match. More precisely, the exemplary embodiment determines the conditional probabilities using a formula, such as

$$P(E_i = \textit{particular state} / M) = a \frac{y}{z} + \frac{1-a}{x} \quad (13)$$



where  $x$  denotes the number of evidence states for evidence variable  $E_i$ ; such as three states for name;  $y$  denotes the number of directory records for which the particular state, for example, exact match occurred;  $z$  denotes total number of attorney match pairs;  $a$  is a smoothing constant, such as 0.999999.

5 Similarly, the exemplary embodiment counts the number of times each evidence state occurs when the candidate name does not match the template name and divides by the total mismatches to get  $P(E_i|\neg M)$  for each state, that is the probability of a particular evidence state match given a mismatch in the names. To reduce computations, some embodiments may sample the matching  
10 and/or mismatching candidate records, for example, by selecting every tenth record.

#### Other Applications

Figure 6 shows that the teachings of the present have applications beyond  
15 facilitating generation of hyperlinks for names. In particular, Figure 6 shows a flowchart 600 of an exemplary method of operating a data retrieval system incorporating teachings of the present invention. Flowchart 600 includes process blocks 610-670.

In block 610, the exemplary method begins with receipt of a query for  
20 information. In the exemplary embodiment, the query has a Boolean or natural-language form and includes a name of an entity, such as a person. In some embodiments, the query is provided by a user of a client computer or access device, such as one of access devices 150 in Figure 1, to a server, such as database server 140 in Figure 1, in a client-server environment. In these cases,  
25 the exemplary method is incorporated into software on a server. However, in other embodiments, the query may be received and processed (in accord with this exemplary method) on the client side before transmission to a server for execution. In such cases, the exemplary method may be incorporated into a browser, a browser add-on, a client-side operating system or searching software.

30 Block 620 entails determining the ambiguity of the query or one or more terms in the query. In the exemplary embodiment, this entails identifying at least

one name in the query and computing a probability of name uniqueness according to

$$P(\text{nameUniqueness}) = \frac{1}{(H \cdot P(\text{name})) + 1} \quad (14)$$

where  $H$  denotes the estimated size of the human population likely to be referenced in the corpus or database targeted by the query. One way to estimate  $H$  is to take the size of the community corpus likely to be referenced and scale it up by some percentage to account for inevitable references to people outside the referenced community.  $P(\text{name})$  is defined as

$$P(\text{name}) = P(\text{first name}) \cdot P(\text{last name}) \quad (15)$$

where  $P(\text{first name})$  denotes the probability of drawing the first name at random from all the first names in a relevant search universe, such as a professional directory and  $P(\text{last name})$  similarly denotes the probability of drawing the last name at random from all the last names in the universe.

Block 630 determines whether to request additional information to assist in answering the query based on the determined ambiguity of a name (or other portion) in the query. In the exemplary embodiment, this entails comparing the computed probability of name uniqueness to a threshold. If the probability of name uniqueness is less than the threshold, execution advances to block 640; otherwise execution continues at block 660.

Block 640 entails obtaining additional information regarding the query. In one exemplary embodiment, obtaining the additional information entails requesting the user for additional information related to one or more ambiguous portions of the query, such as a name in the query. The request, in some embodiments, is presented as a dialog window requesting information related to a profession, a location, and/or an organization associated with the name.

In another embodiment, obtaining additional information entails automatically formulating one or more queries based on one or more ambiguous portions of the received query, such as the identified name or a portion of the name, and executing the query against one or more databases, for example, professional directories or other databases that include names in association with other data. For example, one query can request records or portions of records

having last names that match the last name of a name identified in the received query. The portions of the records, in one embodiment, includes location, organization, and/or professional title information. After obtaining the additional information, execution advances to block 650.

5           Block 650 entails changing the query based on the additional information. In one embodiment that requests additional information from the user, changing the query includes adding one or more portions of the additional information to the query, for example as one or more appended text strings. However, some other embodiments change the query by adding a search  
10 operator, for example, an AND operator, and one or more portions of the additional information. In still other embodiments, particularly some that obtain additional information through use of automatic sub-queries, changing the query includes adding one or more sub-queries, using additional information, such as city, state, organization, and professional title obtained at block 650. In some  
15 other embodiments, changing the query includes changing the scope of the search by for example, adding or deleting one or more target databases for the query, based on the additional information.

Block 660 conducts a search based on the original query or the changed query. In one exemplary embodiment, searching based on the changed query  
20 entails executing the search against an original target database and/or one or more other databases. Execution continues at block 670.

Block 670 entails outputting the results of the queries. In the exemplary embodiment, this entails outputting the results on a display. In one embodiment that changes the received query by adding sub-queries based on the additional  
25 information, outputting the results includes displaying results of the original received query and results of the sub-queries in separate regions of a display screen. In some variants of this embodiment, the results in both regions are ranked.

Still other applications of the teachings of the present invention include  
30 generating new name directories for databases based on famous names, political figures, celebrities, filling gaps in current directories, identifying or discovering gaps in directories. Other applications include automatic generation of dossiers,

and cross-referencing of individuals, businesses, assets, and public and private records.

### Conclusion

5           In furtherance of the art, the inventor has presented various exemplary systems, methods, and software which facilitate logical association of names in documents or other data structures to data structures, such as records, in professional directories or other type databases. Additionally, the inventor has presented various systems, methods, and software for processing and  
10   augmenting queries based on ambiguous query terms, such as entity names.

          The embodiments described above are intended only to illustrate and teach one or more ways of practicing or implementing the present invention, not to restrict its breadth or scope. The actual scope of the invention, which embraces all ways of practicing or implementing the teachings of the invention,  
15   is defined only by the following claims and their equivalents.

**What is claimed is:**

1. A computer-implemented method comprising:  
 identifying a name in a document;  
 determining a rarity indicator for the name; and  
 defining a hyperlink for the name based on the rarity indicator.
2. The computer-implemented method of claim 1, wherein the name is a person name.
3. The computer-implemented method of claim 1, wherein the rarity indicator is a quantity based on a probability of drawing at least a portion of the name at random from a set of sample names representative of a relevant human population.
4. The computer-implemented method of claim 1, wherein the rarity indicator is a quantity based on a size of a human population, a probability of drawing a first portion of the name at random, and a probability of drawing a second portion of the name at random.
5. The computer-implemented method of claim 1, wherein the first portion is a first.name portion of the name and the second portion is a last name portion of the name.
6. The computer-implemented method of claim 1, wherein defining a hyperlink for the name based on the rarity indicator, comprises:  
 identifying one or more non-person-name terms from the document;  
 identifying one or more candidate records in a database based on at least a portion of the name;  
 comparing the non-person-name terms for each of the candidate records to the non-person-name terms from the document;  
 calculating one or more quantities, each based on the rarity indicator for the person name and the comparison of the non-person-name terms for one of the candidate records; and  
 defining the hyperlink based on the one or more calculated quantities.
7. The computer-implemented method of claim 6, wherein calculating one or more quantities, each based on the rarity indicator for the person name and the comparison of the non-person-name terms for one of the candidate records, includes using a Bayesian inference engine.
8. The computer-implemented method of claim 6, wherein defining the hyperlink based on the one or more calculated quantities comprises: comparing the quantities to a threshold; and defining the hyperlink based on a greatest one of the quantities exceeding the threshold.

9. The computer-implemented method of claim 8, wherein defining the hyperlink based on the greatest one of the quantities exceeding the threshold comprises defining a hyperlink to designate the candidate record corresponding to the greatest one of the quantities.
10. A machine-readable medium comprising machine executable instructions for performing the computer-implemented method of claim 1.
11. A machine-readable medium comprising machine executable instructions for performing the computer-implemented method of claim 9.
12. A system for adding a hyperlink to a document including a person name, the system comprising:  
at least one processor;  
a memory coupled to the processor, the memory including instructions for:  
identifying a name in a document;  
determining a rarity indicator for the name; and  
defining a hyperlink for the name based on the rarity indicator.
13. The system of claim 12, wherein the name is a person name.
14. The system of claim 12, wherein the rarity indicator is a quantity based on a probability of drawing at least a portion of the name at random from a set of sample names representative of a relevant human population.
15. The system of claim 12, wherein the rarity indicator is a quantity based on a size of a human population, a probability of drawing a first portion of the name at random, and a probability of drawing a second portion of the name at random from a set of sample names representative of a relevant human population.
16. The system of claim 12, wherein defining a hyperlink for the name based on the rarity indicator, comprises:  
identifying one or more non-person-name terms from the document;  
identifying one or more candidate records in a database based on at least a portion of the name;  
comparing the non-person-name terms for each of the candidate records to the non-person-name terms from the document;

calculating one or more quantities, each based on the rarity indicator for the person name and the comparison of the non-person-name terms for one of the candidate records; and defining the hyperlink based on the one or more calculated quantities.

17. A data structure comprising:
  - a name; and
  - a name-rarity indicator which indicates how likely the name is to refer to more than one entity in a population.
18. The data structure of claim 17, further comprising:
  - one or more organizations co-existent in a document with the name; and
  - one or more locations co-existent in a document with the name.
19. The data structure of claim 2, further including positional information indicating relative position of each organization and each location relative to the name.
20. A computer-implemented method comprising:
  - determining rarity of a name in a population.
21. The computer-implemented method of claim 20, further comprising: associating a hyperlink with the name, with at least a portion of the hyperlink defined based on the determined rarity.
22. The computer-implemented method of claim 20, further comprising:
  - receiving a document including at least one occurrence of the name; and
  - identifying the name in the document before determining its rarity.
23. The computer-implemented method of claim 20, wherein the name is a person name.
24. A system comprising means for determining rarity of a name in a population.
25. The system of claim 24, further comprising:
  - means for associating a hyperlink with the name, with at least a portion of the hyperlink defined based on the determined rarity.

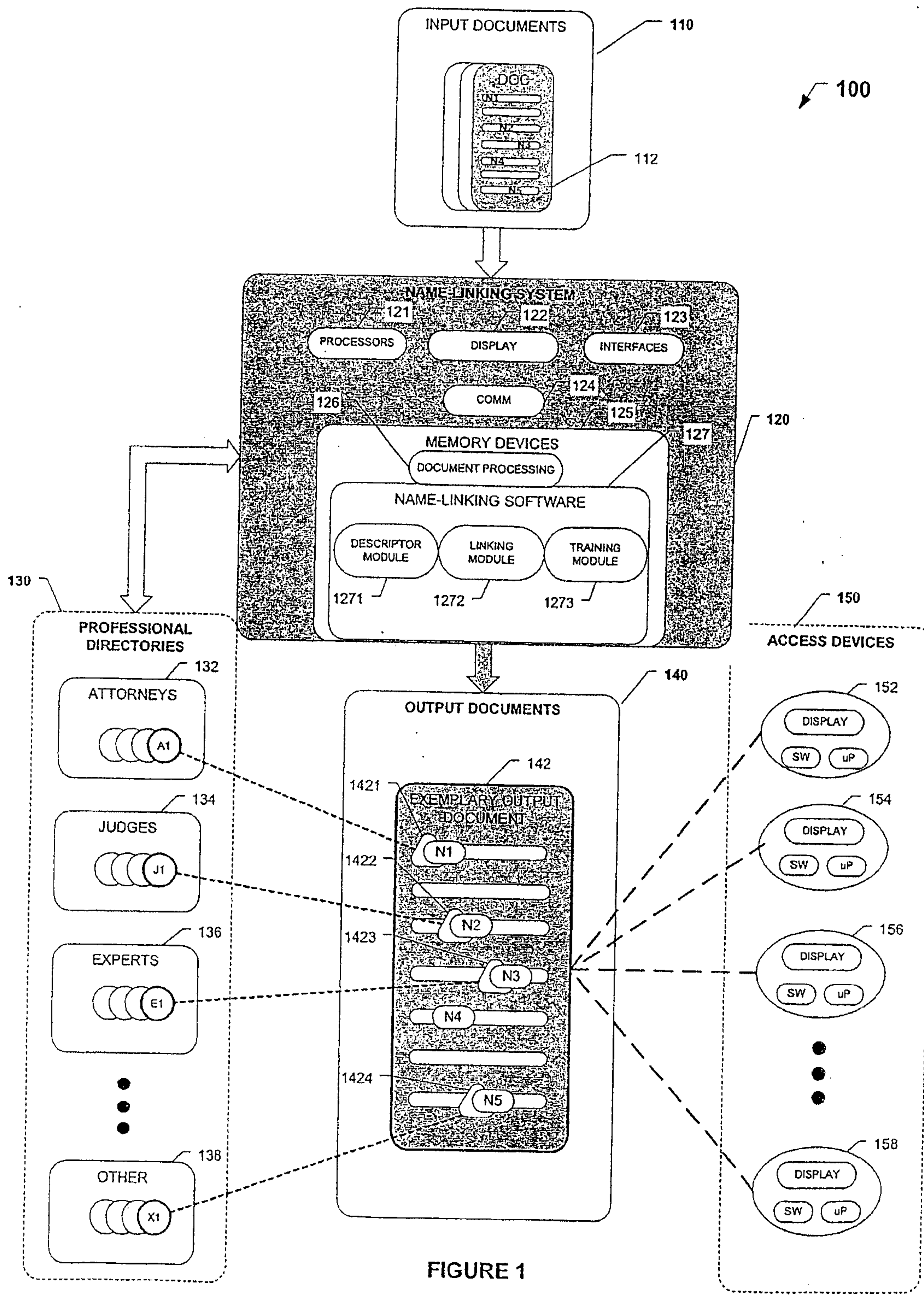


FIGURE 1



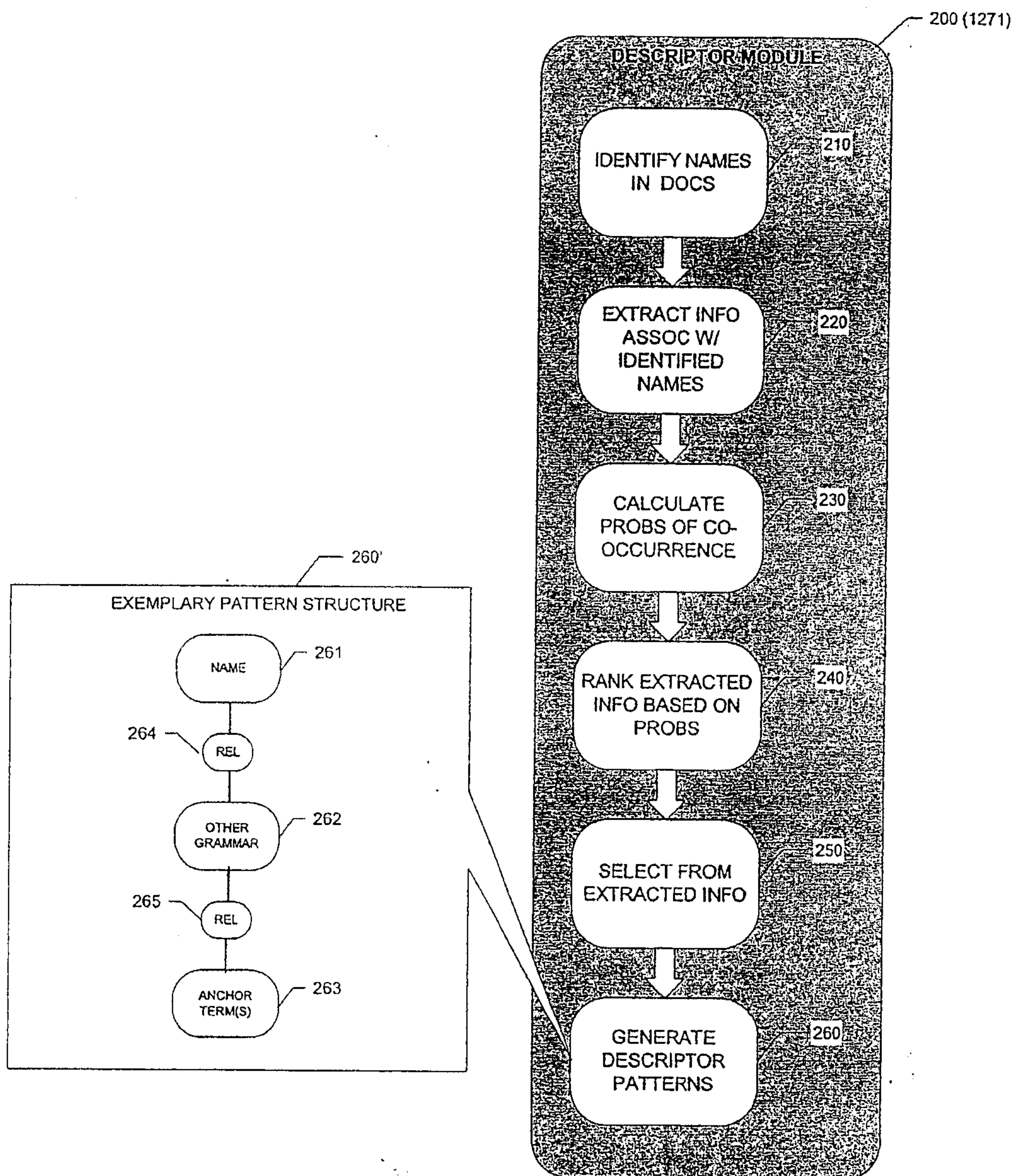


FIGURE 2

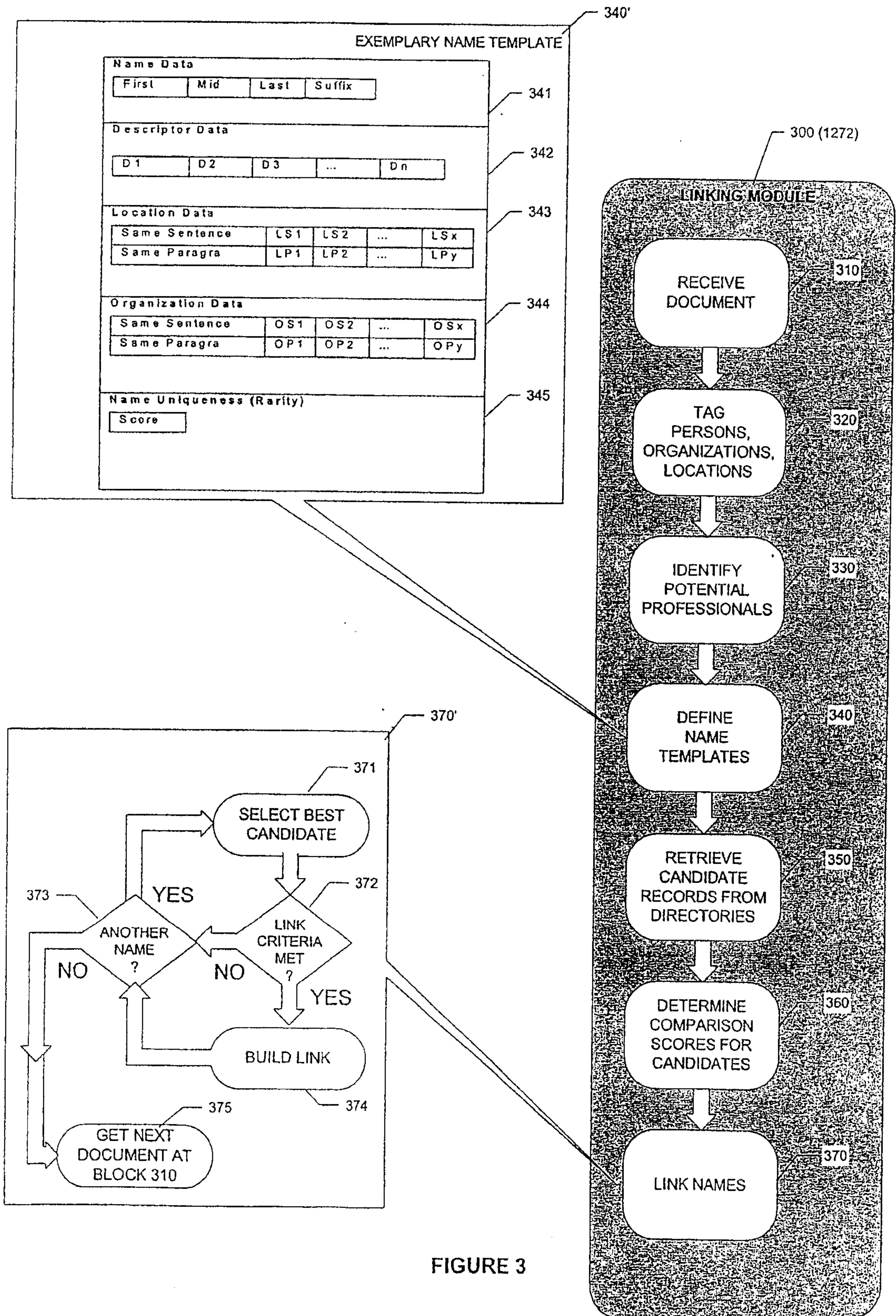


FIGURE 3

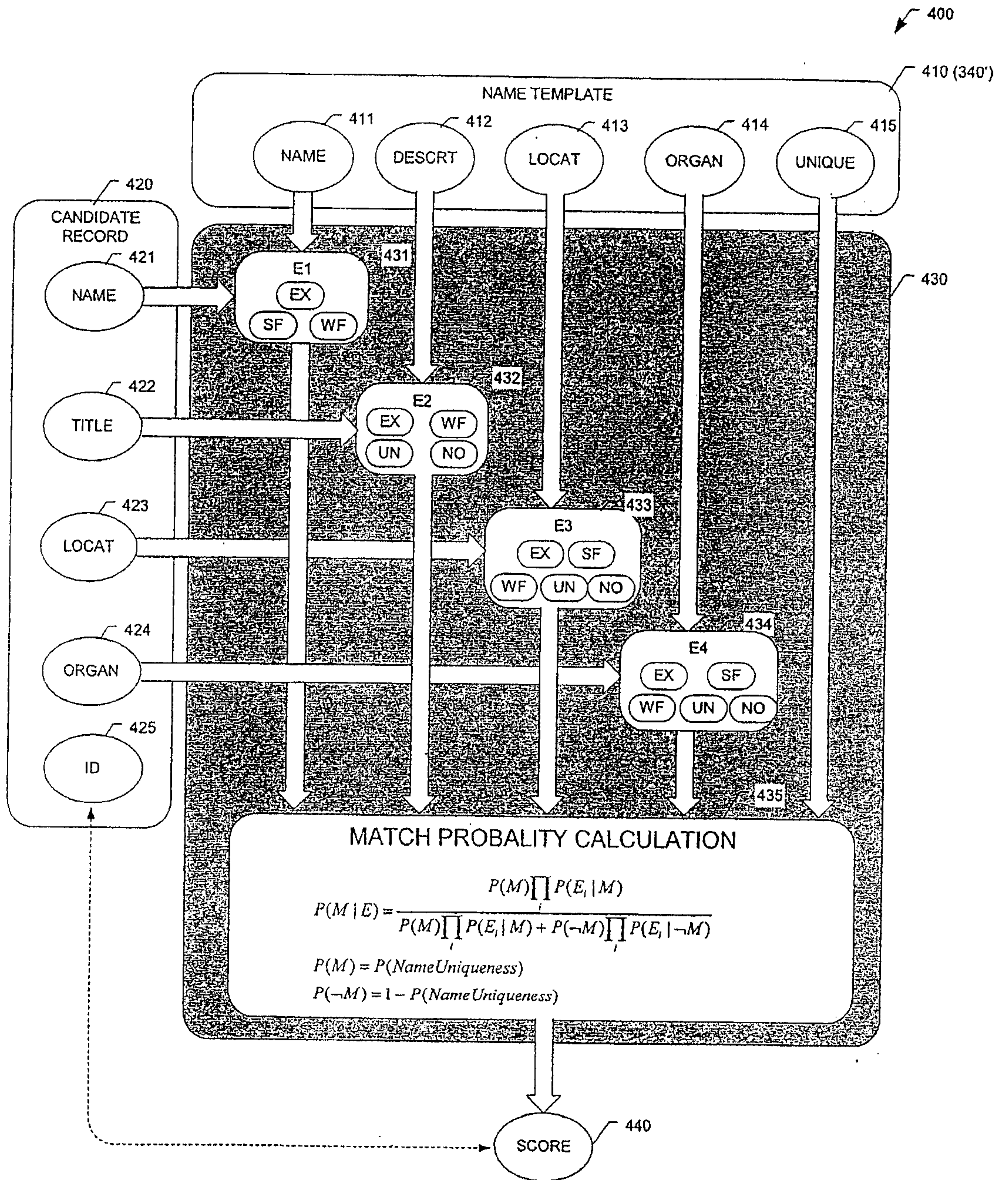


FIGURE 4

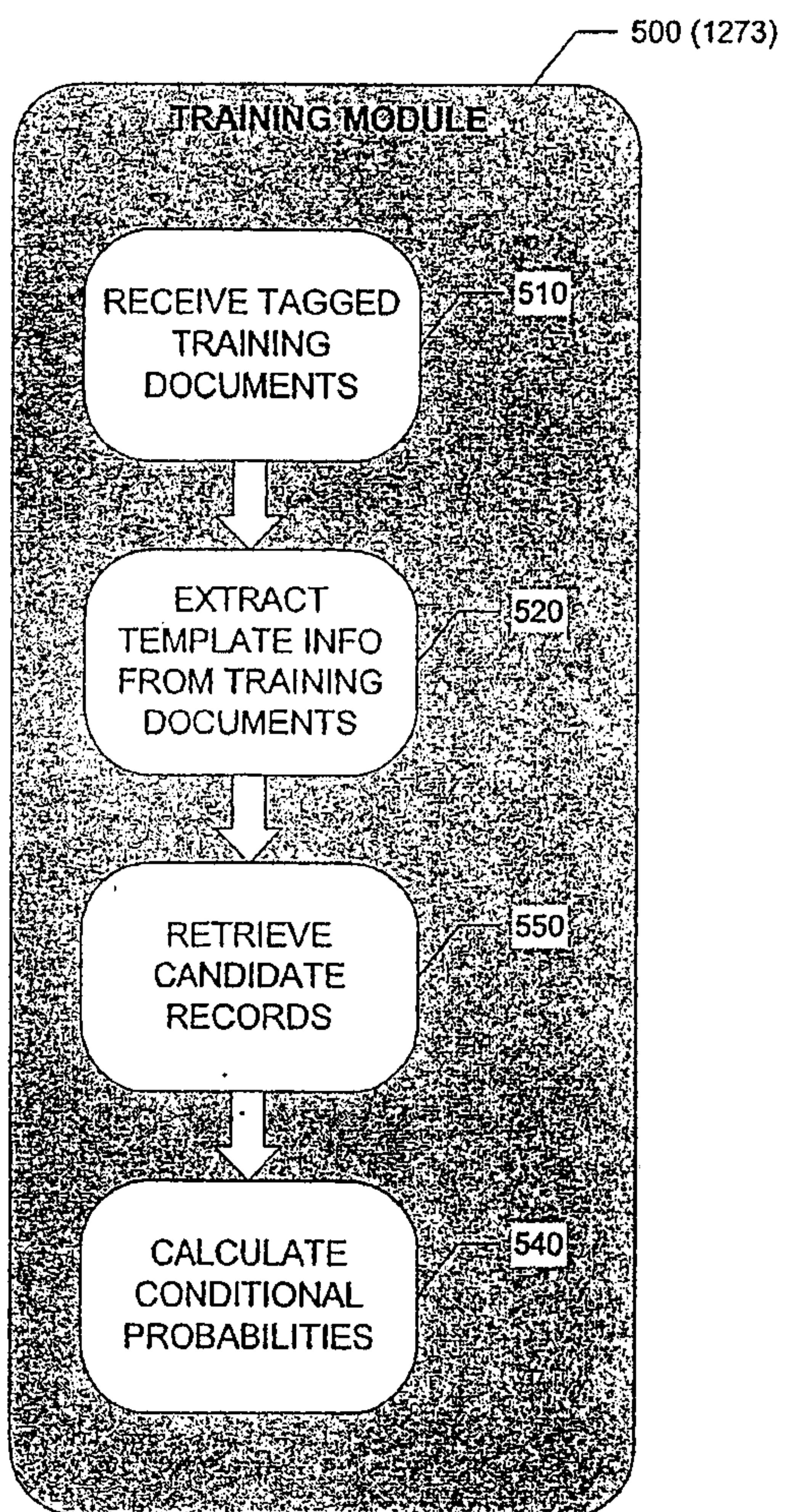


FIGURE 5

6/6

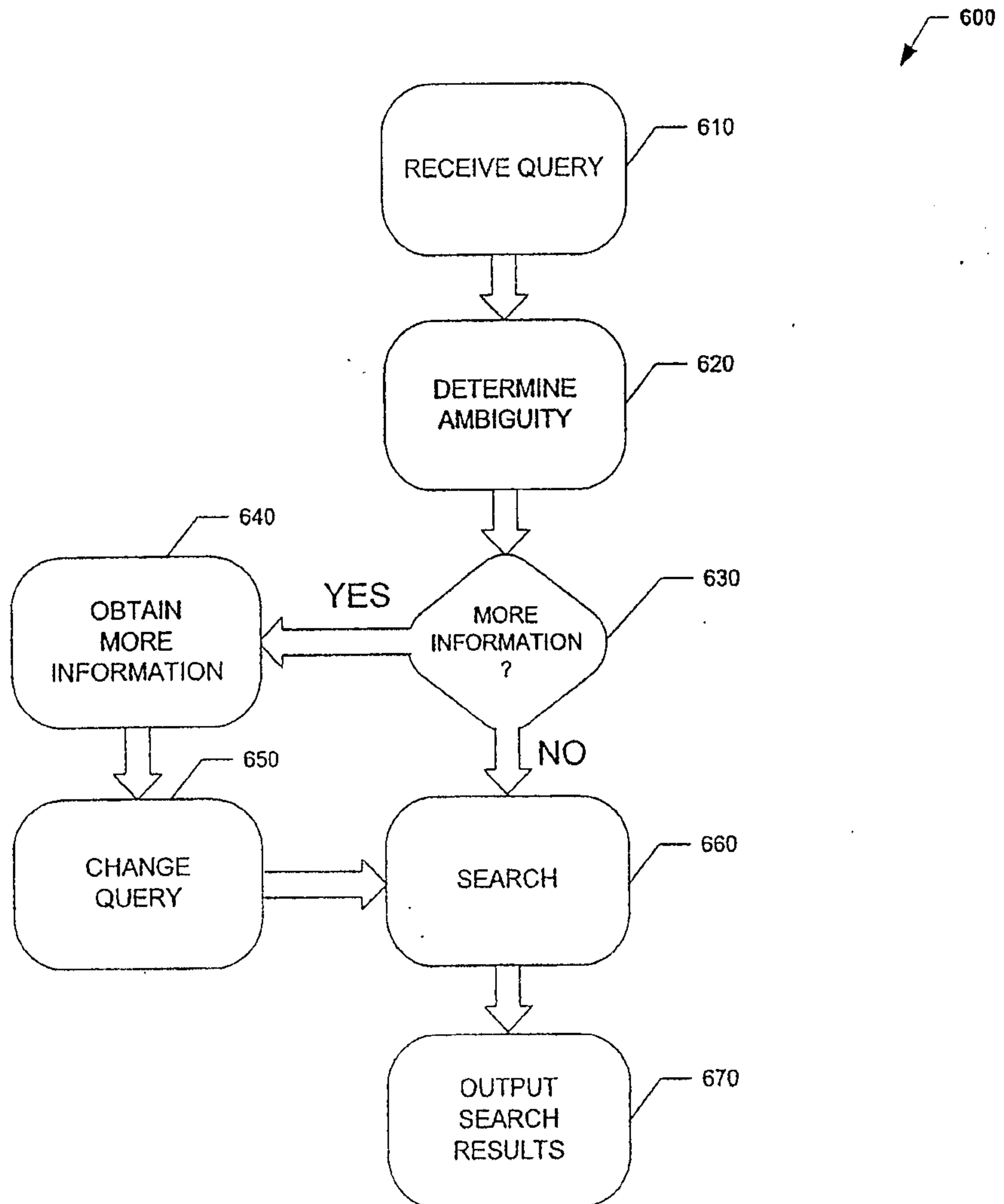


FIGURE 6

